Titre: Title:	Deep Learning Approach for Postprocessing Regularization in Seizure Prediction
Auteur: Author:	Ahmad Chamseddine
Date:	2019
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Chamseddine, A. (2019). Deep Learning Approach for Postprocessing Regularization in Seizure Prediction [Master's thesis, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/4021/

URL de PolyPublie: PolyPublie URL:	https://publications.polymtl.ca/4021/
Directeurs de recherche: Advisors:	Mohamad Sawan
Programme: Program:	génie électrique

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Deep learning approach for postprocessing regularization in seizure prediction

AHMAD CHAMSEDDINE
Département de génie électrique

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
Génie électrique
Août 2019

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Ce mémoire intitulé :

Deep learning approach for postprocessing regularization in seizure prediction

présenté par **Ahmad CHAMSEDDINE**
en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
a été dûment accepté par le jury d'examen constitué de :

Yvon SAVARIA, président
Mohamad SAWAN, membre et directeur de recherche
Jean-Pierre DAVID, membre

DEDICATION

To my wife who was the most caring and supportive,
to my brother Adam Chehouri who was my rock in this journey,
thank you all . . .

ACKNOWLEDGEMENTS

I would like to aknowledge my indebtedness and render my warmest thanks to my supervisor, Professor Mohamad Sawan, who made this work possible. My journey would not have been fruitful without his encouragement and kind advice.

I would also wish to express my gratitude to my colleague Dr. Elie Bou Assi for extended discussions and valuable suggestions which have contributed greatly to my work.

My family, who had never retreat from supporting and encouraging me. My father, who taught me asking the right questions and thinking from outside the box. I would like to thank you all.

This thesis has been written during my stay at the Electrical Engineering Department of École Polytechnique de Montreal. I would like to thank the department and my lab directory for the excellent work condition and the financial support during my studies.

RÉSUMÉ

L'épilepsie est considérée parmi l'une des maladies neurologiques les plus couramment diagnostiquées. Cette condition est caractérisée par l'apparition de crises non provoquées. Près du tiers des patients épileptiques sont pharmaco-résistants. Ainsi, des traitements alternatifs sont considérés, tels que la chirurgie ou la stimulation électrique. Pour atténuer les dommages chroniques de la stimulation électrique programmée, il faut prévoir l'épisode de crise pour déployer sélectivement toute technique préventive. La prévision des crises a été un grand défi pour les neuroscientifiques et les ingénieurs au cours des dernières décennies. Bien que le domaine de l'intelligence artificielle ait connu une percée remarquable au cours des dernières années, la prédiction des crises est toujours difficile en raison de la quantité limitée de données relatives aux patients. De plus, la plupart des résultats des études de prévision des crises ne peuvent être comparés en raison de l'aspect unique du signal d'électroencéphalogramme (EEG) d'un patient. La prévision des crises est basée sur la discrimination de la phase « pré-ictale », qui est une phase transitoire (30-60 min) qui se produit avant l'apparition de la crise. Les classificateurs d'apprentissage automatique sont entraînés à partir de mesures soigneusement choisies du signal EEG et sont optimisées pour distinguer la phase « pré-ictale » de l'activité EEG normale (phase inter-ictale). En raison du rapport signal-bruit élevé de l'enregistrement EEG, la plupart des classificateurs sont enclins à produire de fausses alarmes. Par conséquent, l'étape de régularisation post-traitement est recommandée pour une meilleure optimisation. Les méthodes de régularisation appliquées dans la prévision de crise sont la technique de puissance de tir et le filtre Kalman. Ces méthodes sont, au mieux, des estimateurs quadratiques linéaires. Dans cette étude, nous proposons d'appliquer des méthodes plus personnalisées et spécifiques aux patients; ces méthodes apprennent la meilleure fonction de régularisation uniquement à partir des données. Nous avons prouvé que le réseau neuronal à longue mémoire à court terme (LMCT) peut apprendre une fonction de régularisation optimisée en fonction de chaque individu. Nos modèles ont été formés et testés sur Epilepsy ecosystem database [1].

ABSTRACT

Epilepsy is considered among the most commonly diagnosed neurological diseases. It is a condition characterized by the occurrence of unprovoked seizures. Almost third of epilepsy patients are drug-resistants. Thus, alternative treatments are considered, such as surgery or electrical stimulation. To mitigate the chronic harm of prescheduled electrical stimulation, one needs to forecast the seizure episode to selectively deploy any preventive technique. Seizure forecasting has been a great challenge for neuroscientists and engineers in the last decades. Although the Artificial Intelligence realm has witnessed a remarkable breakthrough in the last few years, seizure prediction is still challenging due to the limited amount of patient data. Additionally, most findings of seizure prediction studies cannot be benchmarked due to the patient-specific aspect of electroencephalogram (EEG) signal. Seizure prediction is based on discriminating the preictal phase, which is a transitional phase (30-60 min) that occurs prior to the seizure onset. Machine learning classifiers are trained on carefully selected measures of the EEG signal, and optimized to distinguish the preictal phase from the normal EEG activity (interictal phase). As a result of the high signal to noise ratio (SNR) in the EEG recording, most classifiers are prone to produce false alarms. Therefore, post-processing regularization step is recommended for a better optimization. The regularization methods applied in seizure prediction are firing power technique and Kalman filter. These methods are, at best, a linear quadratic estimators. In this study, we proposed applying more customized and patient-specific methods that learn the best regularization function purely from data. We proved that Long Short-Term Memory (LSTM) neural network can learn an optimized regularization function based on each individual. Our models were trained and tested on Epilepsy ecosystem database [1].

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

List of Algorithms

LIST OF SYMBOLS AND ACRONYMS

DL	Deep Learning
ML	Machine Learning
SS	sensitivity
SP	speicificity
NN	Neural Network
AUC-ROC	Area Under Curve Receiver Operating Characteristics
ANN	Artifical Neural Networks
SVM	Support Vector Machine
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
LSTM	Long-Shot Term Memory
BLSTM	Bidirectional Long Short Term Memory
KF	Kalman Filter
FP	Firing Power
GA	Genetic Algorithm

CHAPTER 1 INTRODUCTION

1.1 Epilepsy

1.1.1 Definition, epidemiology and etiology

Epilepsy is one of the most frequently occurring neurological disorders, affecting over 50 million people worldwide and causing significant morbidity and mortality [6] [7]. It is an abnormal condition that consists of unprovoked repetitive seizures. A single event or a status epilepticus (SE) is equivalent to multiple episodes of seizures occurring in 24 hours. Patients with one single unprovoked seizure (absence of precipitating factors), febrile seizure, neonatal seizure (seizures that occur in the first month of life) or seizures occurring with acute systemic illness (infection, intoxication, etc.) are excluded from epilepsy [8]. Epilepsy can occur at any age. Nevertheless, it has a higher incidence at the pediatric and the geriatric age (younger than 12 and elder than 65) [9]. The age-adjusted rate ranged from 16 to 51 per 100,000, and 20–66% of incident epilepsies are epilepsy with partial seizures [6]. Epilepsy can be a result of genetic predisposition, gliosis from acute brain injury (infection, intoxication, ischemia, etc.), angiomas, or degenerative disorders.

1.1.2 Epileptic seizures

A seizure is defined as a group of neurological symptoms which come as a result of an abnormal synchronous electrical activity of a group of neurons [10]. If the abnormal electrical activities are restricted to a limited area of the brain, it is called a focal seizure. If it involves both hemispheres, it is called a generalized seizure. Focal seizures can occur in frontal, temporal, parietal and occipital lobes [11]. In the case of generalized seizures, the widespread electrical discharge causes impairment of the consciousness. However, the clinical presentation of the focal seizure is dependant on the location of the seizure activity. For instance, If the seizure onset is in the occipital lobe, it is expected to show visual symptoms. Generalized seizures can be subdivided based on the symptoms the patient is showing. Absence seizures are presented with complete unconsciousness that lasts for a short time (less than 30 seconds) [12]. Myoclonic seizures show an unintentional contraction of muscles that lead to an uncontrolled movement for a concise period. The Clonic seizure is similar to the myoclonic seizure, except the contraction is repeated every 2 to 3 seconds. A Tonic-Clonic seizure is a Clonic seizure preceded by a strong contraction of muscles. The Atonic seizure causes a loss of muscle contraction. Focal seizures are susceptible to spread to the bilateral hemisphere

and generate Tonic-Clonic convulsions [13].

1.1.3 Electroencephalography

Electroencephalogram (EEG) is a measurement tool performed by applying electrodes on the skull of the patient. It displays the fluctuations in the electrical activity of the brain, which are generated by the extracellular field potentials. The final output represents the changes in voltage measured by electrodes as a function of time. The analog measurements are converted to a digital signal with a high time resolution (can reach an order of μs), which provide rich information to study the dynamical changes in the electrical activity of the brain. EEG measures the sum of the activities, more precisely the extracellular post-synaptic potentials, of a large group of neurons. EEG (especially scalp-recorded) is designed to read the activity of a dipole (a field with negative and positive poles). Therefore, neuronal cells need to be positioned parallel to each other to measure its activity on the surface of the brain. This condition is mainly applied on Pyramidal cortical neurons which are perpendicularly oriented to the surface. Thus, they are the major contributor in the scalp-recorded EEG signal [14]. Scalp-recorded EEG is highly sensitive to the distance and the angle between the electrode and the dipole. Thus, the EEG signal is affected by the resistivity of the skull. Additionally, the skull-brain surface has a high inhomogeneity and anisotropy, all of which can make scalp EEG highly attenuated and distorted.

The number of electrodes placed on the skull controls the spatial resolution of the EEG recordings. The more electrodes we apply, the higher spatial resolution we obtain. In medical practice, the diagnosis of neurological diseases can be achieved with less than 32 electrodes, while in research, a high spatial resolution is required where up to 256 electrodes can be placed. This latter is mainly referred to as high-density EEG [15].

The positioning of the electrodes follows different approaches (10-20 systems, 10-10 system, and 10-5 system). Essentially, each electrode is labeled by an upper case character and a digit. The character designates the region on the scalp and the digit indicates the specific location in each region. F, C, O, A, T, and P stand for frontal, central, occipital, auricular, temporal and parietal. The 10-20 systems (Figure 1.1), also called the international system, is considered the standard system. The 10-10 and the 10-5 systems are extensions of the standard system and are applied for higher density measurements, which is only required in research [16].

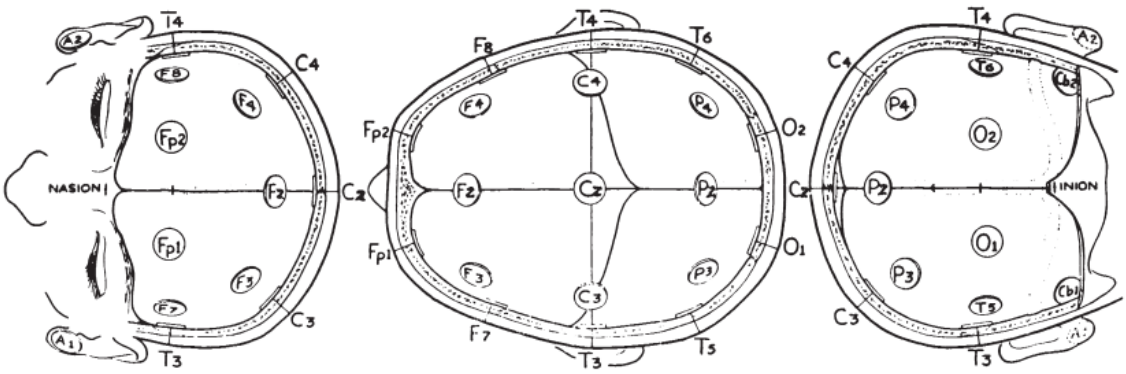

Figure 1.1 Scalp EEG electrode placement according to the 10-20 international electrode placement system [2](reproduced with permission)

1.1.3.1 Intracranial EEG

Intracranial EEG (iEEG) averts the resistivity of the skull and collects a stronger signal with minimal distortion. It is an invasive method that consists of implanting the electrodes directly inside the skull. IEEG has a higher spatial resolution compared to a high-density scalp EEG. It can reach a spatial resolution up to one electrod per millimeter. It is recorded using a grid or a strip of electrodes placed directly on the brain. Unlike scalp EEG, iEEG does not follow a standard positioning system (Figure 1.2, and it has a patient specific positioning).

1.1.3.2 Bands of frequency

Typically, EEG describes the rhythmic activity of the brain, which is divided into bands of frequency. Each band of frequency has its specific spatial distribution over the scalp and has been correlated with a particular state of mind. The Delta rhythm (< 4 Hz) has a high amplitude signal, and it is highly recorded at sleep in the frontal lobe in adults and posteriorly in babies. The Theta rhythm (4-8 Hz) is active more in juveniles and associated with drowsiness in adults and teens, and it is characterized by its irregularity. The Alpha rhythm (8-15 Hz) is focused bilaterally on the posterior part of the brain, and it is associated with reflecting/relaxing activities. The Beta rhythm (16–31 Hz) has a low amplitude wave that is active in both sides of the brain, mainly in the front side. It is associated with active thinking, anxiousness and high alertness. The Gamma rhythm (>32 Hz) is more recorded in the somato-sensory cortex and is correlated with cognitive activities (e.g., data processing,cross-sensed perceptions).

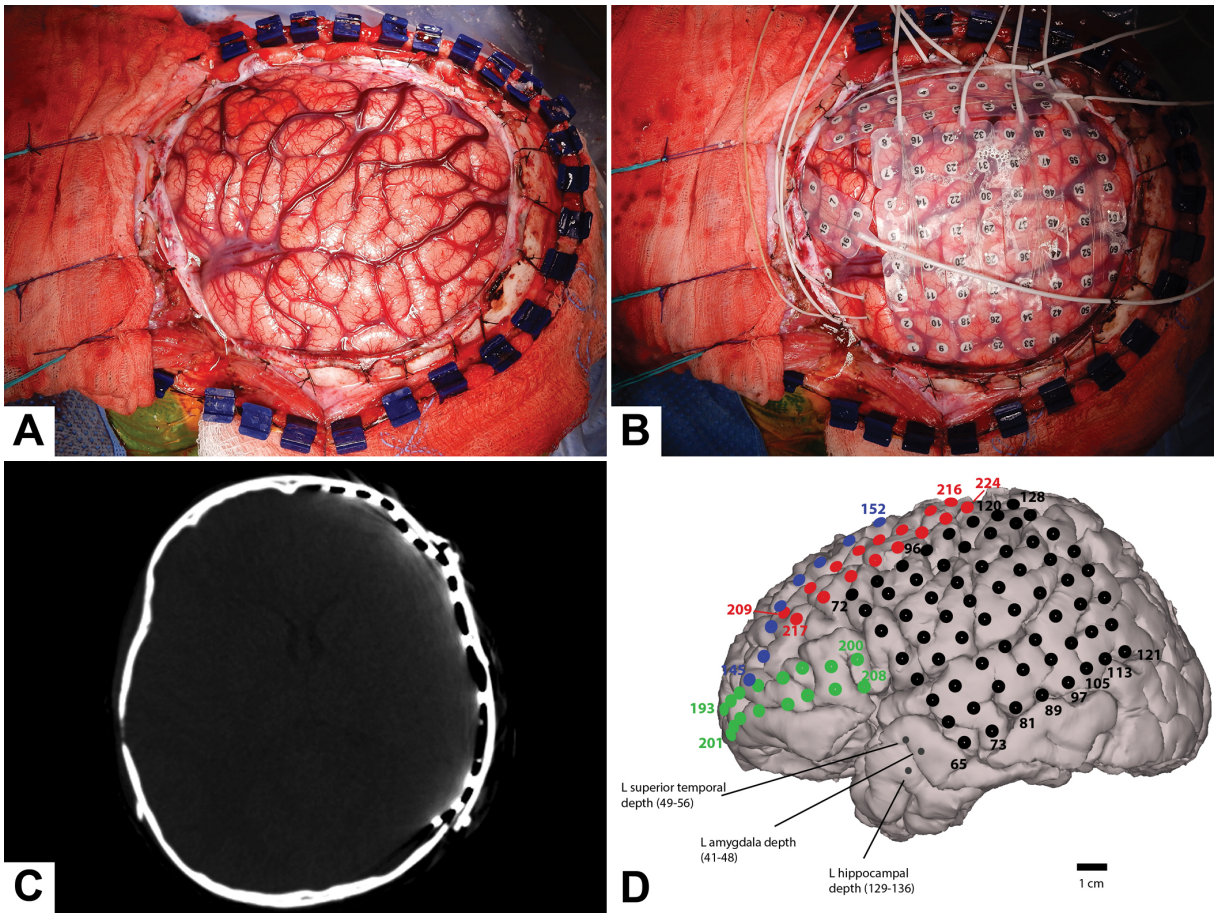

Figure 1.2 Placement of intracranial grid. A: Photograph showing a left-sided craniotomy. B: Placement of surface and depth grid of electrodes secured to the dural cuff. C: Axial CT scan exhibiting the elevated placement of the bone flap. D: A postoperative image showing the locations of surface grid and the entry points for depth electrodes. [3] (reproduced with permission)

1.1.3.3 EEG in Epilepsy

In addition to the rhythmic activities, EEG can extract transient activities in the brain, which is essential to detect abnormal activities such as epileptic seizures. Focal epilepsy is characterized by focal epileptogenic zones, which is exhibited by EEG through two types of transient activities: 1) interictal epileptiform discharges (IEDs) (20-200ms) and 2) ictal discharges. IEDs are asymptomatic and they have a high contrast, thus, easy to distinguish from background. IEDs appear in 1-13% of normal individuals, while in tonic-clonic generalized seizures, it has been detected in 50% of cases [17].

Ictal discharges are dependent on the type of seizures. In focal seizures, they tend to have a specific pattern with respect to spatial distribution and amplitude. However, in general-

ized seizures, a sudden high amplitude discharge evolve to all the areas of the brain. Ictal discharges are often presented with epileptic clinical manifestations.

1.1.3.4 EEG limitations

EEG is an effective tool for diagnosis and research of epilepsy and is characterized by a high temporal resolution and acceptable spatial resolution (especially IEEG). Nevertheless, EEG signals are highly vulnerable to artifacts and noise [18]. Artifacts can be biological which is caused by the voluntary and involuntary muscle movement by the patient. The frequency of biological noises are often outside the frequency range of brain activity. Low frequency components (< 1 Hz) are caused by sweating, while frequencies higher than 100 Hz are related to muscle electrical activities. Applying low pass and high pass filters is considered the primary solution for such artifacts. Artifacts can also be a result of environmental factors such as the devices themselves, bad connections of the electrodes, and the magnetic fields from the electrical current which creates a notch shaped noise (50 Hz in Europe and 60 Hz in North America). Noises produced by the environment can be mitigated by isolating the patient and using a conducting gel to enhance the electrodes connectivity. Similar to biological noises, a notch filter can easily remove the artifact caused by the 50/60 Hz buzz. Additionally, the EEG signal represents a final output of the superposition of different independent signals resulted by different neuronal groups. One can apply Independent Component Analysis (ICA) [19] algorithms to guarantee a disentangled reading of the brain activity. However, ICA techniques require a lot of computation and cannot be applied in real time.

1.1.4 Treatment of Epilepsy

The final ideal goal of treating epilepsy is to reach a point of no seizure and no side effects. Practically, we aim to reduce or eliminate the risk of seizure recurrence with minimal side effects. Thus, obtaining optimum results with minimal side effects can be considered as a constrained optimization problem. Treatment can be either medical, which is based on Anti-epilepsy Drugs (AED), or non-medical.

1.1.4.1 Medical treatment

Medical treatment is considered the pillar of epilepsy treatment. However, there is a wide range of AEDs and their efficiency is highly relative to the type and underlying causes of epilepsy and the medical profile of each patient. Any Medical treatment strategy should imply answers for the three following questions: "When to start?", "When to stop?", and "What is

the type of medication?". Patients with the first occurrence of unprovoked seizure have 21 to 45 % risk of recurrence in the next year [20]. Thus, immediate AED treatment reduces the risk of recurrence of epilepsy in the next two years by 35% [21]; however, the possibility of a long-term seizure remission remains high. As for the discontinuation of AEDs, it is recommended to wait at least two years of seizure-freedom before discontinuation in children and 2-5 years in adults [20]. Medication can be either a broad spectrum or narrow spectrum AEDs. Broad spectrum AEDs like Lamotrigine, clonazepam, phenobarbital, topiramate, valproic, levetiracetam, rufinamide, topiramate, zonisamide, perampanel, and clobazam are used for the treatment of all types of seizure.

As mentioned above, AED selection is highly relative to the underlying causes of epilepsy and the profile of the patient. Some AED, like valproic acid, are considered gold standard treatments; nevertheless, they can cause dangerous side effects for certain patients (e.g., high risk of teratogenesis in pregnant women). In case of unbearable side effects, a patient can transfer to a narrow spectrum AEDs like phenytoin, carbamazepine, clonazepam, gabapentin, and lacosamide. Although the medical treatment is successful for most patients, AEDs are known to have two limitations: 1- For some patients, the side effects are unbearable which lead to drug discontinuation; 2- Almost a third of epilepsy patients have resistance to AEDs. Technically, epilepsy is resistant to AEDs in all patients since AEDs are palliative treatment and do not treat the underlying pathological causes of the seizures. Practically, drug-resistant epilepsy is considered when 12 months of treatment fails to achieve seizure freedom. A large study [22] has shown that 36% of patients, who had never been treated for epilepsy, do not achieve seizure freedom after one year treatment. Additionally, AEDs have two critical limitations: 1-They may lead to seizure aggravation often called Idiopathic generalized epilepsies (IGEs); 2- Drug tolerance, where the body develops an adaptive behavior toward AEDs.

1.1.4.2 Surgical treatment

Surgical treatment is considered in patients with drug-resistant epilepsy, especially when seizures occur frequently or severely (increasing the mortality rate or impairing the quality of life). In general, resective surgery is more efficient in patients with a consistent location of the epileptic focus and it is best considered in patients with an epileptic lesion in the temporal lobe. However, obtaining an accurate location of the epileptic focus is quite challenging and requires multiple monitoring tools [23]. It involves a combination of accurate medical history, video-EEG monitoring, MRI and positron emission tomography which serve in studying the origin and the spatial distribution of the seizure epilepsy.

1.1.4.3 *Electrical stimulation*

Electrical stimulation has also been considered an effective alternative to medical treatment, especially when the focus of epilepsy is surgically inaccessible. Electrical stimulation can be classified based on the location of the stimulus or based on the technique. There is three followed approaches in electrical stimulation: 1- Vagus Nerve Stimulation (VNS) which is valid as an alternative treatment for medically intractable partial onset seizure patients older than 12 years of age [24]; 2- Cortical stimulation (CS) which is an approved adjunctive option for patients with refractory focal epilepsy and a well-located seizure focus [25]; 3- Deep brain stimulation (DBS), which targets the centromedian and anterior thalamic nuclei, the subthalamic nucleus, the cerebellum, hippocampus and caudate. As for the electrical stimulation technique, one can apply open loop stimulation or closed loop stimulation. The open loop method (mainly applied in VNS and DBS) consists of pre-scheduled stimulation independent of the brain electrical and physiological activity. However, the closed loop method (applied in CS) is a responsive stimulation for brain electrical activity. Its primary goal is to prevent or shorten the seizure episode. Subdural implanted electrodes continuously monitor the brain activity, and upon abnormal activity electrical stimulation is delivered to hinder any potential incident.

1.2 Problem statement

Seizure epilepsy is been addressed as a disease, while it is technically an abnormal manifestation of different diseases. As mentioned in the previous section, the medical treatment of epilepsy is a palliative treatment that has a wide range of side effects. Patients with intractable epilepsy are unable to benefit from AEDs fully and suffer from a deteriorated lifestyle. This latter brings a lot of social and economic burden to patients and their support system. Promisingly, electrical stimulation carries potential in controlling or preventing seizure. Open loop electrical stimulation is a proven adjunctive treatment for patients with drug-resistant epilepsy. However, any treatment strategy needs to balance between its therapeutic impact and undesired side effects. Open loop electrical stimulation, as a chronic and scheduled stimulation, can yield to permanent brain lesions. On the other hand, closed-loop electrical stimulation is less invasive and selectively stimulate epilepsy focusing upon abnormal brain activity. Nevertheless, this technique raises the need for accurate and automated estimation of abnormal electrical stimulation. Closed-loop electrical stimulation requires 24 hours of brain activity monitoring, therefore, EEG is the most practical tool to record brain activity. However, the EEG signal is prone to noises and artifacts and it represents a superposition of different electrical neural activities.

The success of closed-loop electrical stimulation is highly dependant on the accuracy of detection or prediction of seizure episodes. Thus, automating the process of detecting/predicting has become an engineering problem. The aim is to find and develop algorithms that discriminate regular EEG activity from any abnormal behavior. The challenge is in finding the best set of discriminative features and the best classification in term of accuracy and computation complexity. Besides, it is crucial to handle the random nature of the EEG signal, due to its high susceptibility to noises, and artifacts.

1.2.0.1 From seizure detection to seizure prediction

Automatic seizure detection has witnessed a remarkable advance in the last two decades [26]. This has been a result of the progress in computation and embedded systems. Detecting the onset of a seizure episode is possible by simple visual inspection. It does not require complex features to be discriminated. One can notice the sudden changes in the EEG amplitude and frequency in one or multiple electrodes. Thus, the primary challenge has always been in computing the discrimination function with minimum latency to deliver electrical stimulation with minimum intervention time. Seizure detection techniques state of the art is achieving very sensitive and specific results. Nonetheless, in its best results, the detection is done after the seizure onset which is often too late to be controlled and the clinical manifestation are already developed.

Many research findings have been showing that brain electrical activity emerges gradually into a transition phase before reaching the ictal phase [27]. The transition phase is often called "preictal", which is an elusive phase (hard to strictly define) and mostly impossible to visually distinguish it from the interictal phase. Studies have shown that statistically significant changes in linear and non-linear features are detected in the preictal phase [27] [28]. Thus, it is possible to discriminate the preictal phase from the interictal phase. Shifting the target from detecting seizure to detecting the preictal phase is equivalent to predicting the ictal phase. The primary challenge remains in finding the best set of features that discriminate preictal from the interictal phase. This challenge is coupled with multiple constraints and limitations: 1- The features need to be computed in real-time; thus a balance between the number of features and its computational complexity needs to be addressed; 2- EEG signal pattern is patient-specific; therefore, an algorithm needs to be customized to each patient; and 3- The seizure electrical behavior is pseudo-random and relies on the underlying triggering causes. This adds obstacles for any algorithm to achieve high accuracy.

1.3 Research objectives

The overarching goal of this thesis is to improve the existing state of the art of seizure prediction. As mentioned above, seizure epilepsy has a very complex mechanism and it's behavior is irregular. Moreover, the main tool for seizure prediction, EEG, is prone to noises and only detects the result of neural signal superposition. The mission of prediction requires a significant amount of data and minimum spatial resolution (not less than 16 electrodes). Yet, with the presence of a big dataset and the best combination of selected features and classifiers, it is highly possible that the model generates false alarms. One should apply post-classification regularization to decrease the number of false alarms. Regularization techniques are mainly simple linear averaging function or at most, a linear quadratic estimation based on a Kernel filter. The first method takes a decision binary number and counts the number of firing alarms. If the number of firing is greater than an arbitrary threshold, then an alarm is generated. This technique exhibits a good memory and reflects the temporal dynamics. However, its function is very simple and static. Additionally, this technique takes the binary output of the classifier, which causes a big loss of information. Kalman filters, on the other hand, are relatively adaptive and able to estimate a filtering function based on the prior sequence points. Nevertheless, it is not a complete machine learning approach, and it has limited memory.

1.3.1 Contributions

Our specific aim is to prove that the regularization step can be learned from data using recurrent deep artificial neural networks and if enough data is provided, the optimized learning function can achieve higher results than ones in the current state of the art. Recurrent ANNs are able to learn a long term memory and is fully optimized by the dataset, which grants it the advantages from the aforementioned techniques. We propose a deep learning-based approach to independently learn a regularization function that outperforms classical methods in specificity. Our method was also able to increase the sensitivity of the baseline classifier, which extends the objective of regularization beyond false alarm suppression. Seizure prediction literature emphasizes the importance of post-processing regularization. Nevertheless, the methods followed are limited to Kalman Filters and Firing Power technique. Based on our extensive review, our work is the first attempt to apply a function trained from data to regularize the classifier output.

1.3.2 Overview of the structure

This master thesis is organized as follows: Chapter 2 presents the state of the art of seizure prediction and review the previous works in each block of the general seizure prediction pipeline. Next, we discuss the limitations and the future prospects of the current state. Chapter 3 provides a theoretical introduction and definition of methods and algorithms implied in our experiment. It also presents the methodology followed in our study. Chapter 4 exhibits our comparative results and highlight the main findings that endorse our hypothesis. Finally, a conclusion of our master thesis.

CHAPTER 2 LITERATURE REVIEW

Seizure prediction has been studied by a growing community for the last three decades [4] [29] (Figure 2.1). The seizure prediction paradigm has shifted throughout the years towards an algorithmic and automated approach. Many review papers of seizure prediction have been published [4] [28] [30] [31]. Considering the scope of our master thesis, we will be more focusing on reviewing the algorithmic part, especially, the different classifications and regularization approaches. Various algorithms have been suggested for solving the seizure prediction problem; nevertheless, they are all considered variations of a general approach that will be reviewed in this chapter in a block-by-block fashion. In each block, we will present the prominent works and contributions. Finally, we will discuss the latest advances and what contributions are needed to make the state of seizure prediction medically and computationally practical.

2.1 State-of-the-art

Typically, the algorithms that aim to detect the preictal phase, by processing the EEG signal, apply a pre-processing step that yields a better signal to noise ratio. Next, they extract linear and non-linear features that are able to discriminate preictal phases from interictal ones. The classification step needs to be applied with minimum time and computation complexity. Therefore, a features' selection step is recommended, especially if the features' dimensional space is challenging for the discriminator. Next, a classifier is trained based on supervised machine learning. Hence, labeled data sample from both preictal and interictal state are required. Figure 2.2 presents a concise description of the main algorithm. Classifiers, despite being trained, are prone to produce false alarms so a post-processing regularization step is recommended.

2.1.1 Data acquisition

Seizure prediction algorithm is, essentially, a supervised machine learning method to learn the best discrimination function for the EEG signal. It is crucial to collect datasets that reflect the real distribution of the selected features set. EEG signal can be either scalp-recorded or intracranially placed. As elaborated in the previous chapter, scalp-recorded EEG consists of placing equally spaced electrodes on the scalp which makes it more sensitive to environmental noises. However, iEEG is invasively placed on the brain cortex as a grid of electrodes. Studies

have considered both types of EEG signals for seizure prediction.

Rasekhi et al. [32] studied the preprocessing effects of 22 univariate linear features on the accuracy of seizure prediction methods. Their findings showed that scalp-EEG (sensitivity=76.67%, FPR=0.08 h-1) performance was better than that of iEEG (sensitivity=68.7%, FPR=0.33 h-1). However, the number of patients in their experiment was relatively low and statistically insignificant. In a more extensive study, Teixeira et al. [33] applied a subject-specific algorithm on 227 patients with scalp-EEG recording, and 42 patients with iEEG recording. Their study showed a better performance for scalp-EEG; however, the Kruskal-Wallis (K-W) test (p=0.01) showed that the difference was insignificant. Similarly, in a comparative study of scalp-EEG and iEEG, Bandarabadi et al. [34] showed that performance with scalp-EEG recording is equally good to the performance with iEEG recording.

In recent years, multiple databases have been created such as the Epileptologie from the

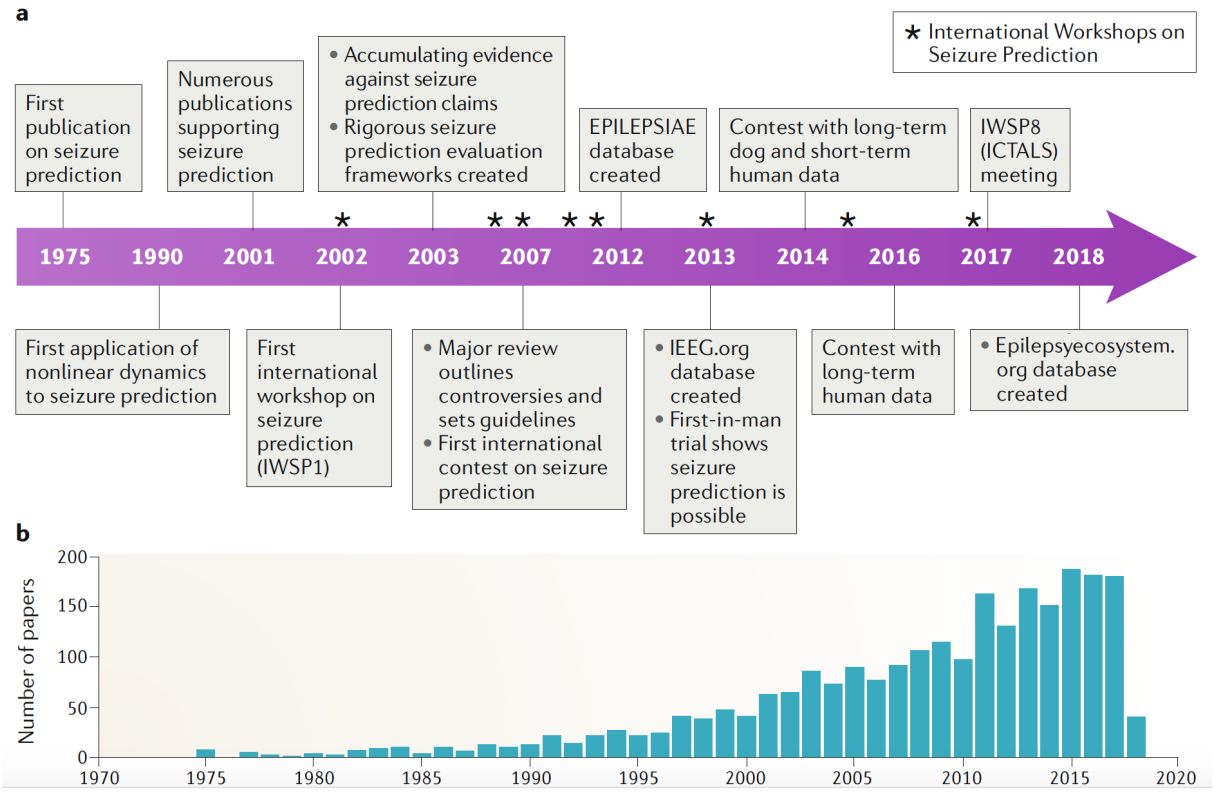

Figure 2.1 The growth of seizure prediction research community [4]. a: A timeline showing the evolution of seizure prediction concept and the main events in this field. b: The number of published paper on seizure prediction in the last 30 years. Sourced by PubMed using the keywords "seizure anticipation", "seizure forecasting" and "seizure prediction" (reproduced with permission)

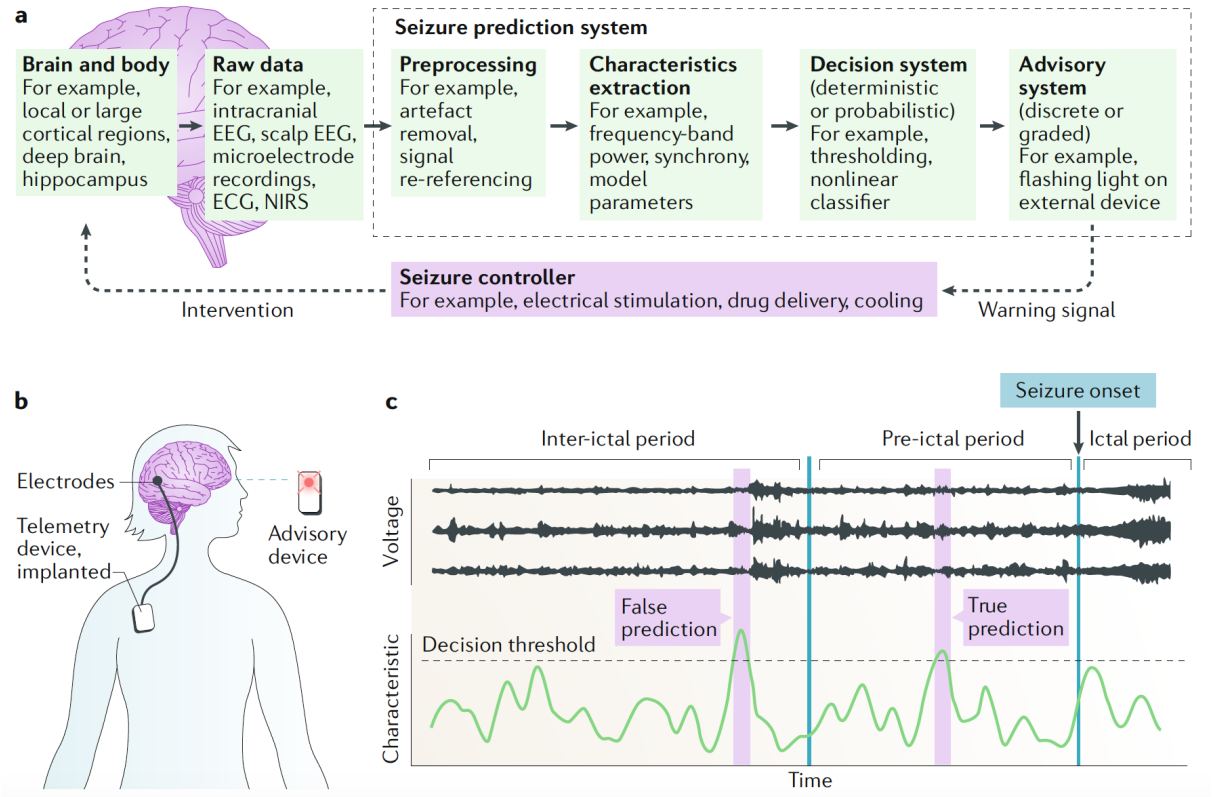

Figure 2.2 Block-diagram for the seizure prediction technology [4]. a: Data acquisition using different recording techniques, pre-processing the data before extracting features, and training a classification technique ; b: A portable seizure alert system that implements closed loop seizure prediction approach in part a. IEEG is recorded by placing electrodes on the surface of brain cortex. Signals are transmitted to an external processing device using a telemetry unit; c: A simple demonstration of seizure prediction concept. The prediction problem has became a detection of the preictal phase. A classifier with a given discriminative threshold projects a set features (multi-dimentional) to a scalar number that represents the probability of a preictal state. During an interictal phase, if the probability is greater than the threshold then we have a false alarm. If the same occurs during a preictal phase, we have a true prediction (reproduced with permission)

University of Bonn and the epilepsy database of Boston Children's Hospital and the University of Freiburg. Up to this time, the largest seizure prediction database is the European Database on epilepsy, EPILEPSIAE [35]. It provides recordings for 250 patients and a total of 45,000 h of EEG data and 20% of patients underwent iEEG. Nevertheless, the seizure prediction problem is patient-specific; thus, the number of seizures recorded per patient is more important than the total hours of EEG and the total number of patients. Recently, a new database was created by Melbourne-University, Epilepsy Ecosystem [1], providing a 10608 h, on average, of iEEG recording per patient with 16 channels sampled at 400 Hz. Although

the database has records for only three patients, all of them had the lowest results in the Cook et al. trial [36]. Hence, any algorithm, trained on these patients, has the potential to be extrapolated to other individuals. Table 2.1 summarizes the specifications of different databases in terms of number of patients, number of seizures, and the total recording hours. It is important to note that an additional seizure prediction database has been created to investigate seizure prediction feasibility in dogs. Howbert et al. [37] studied the possibility of seizure prediction of naturally-occurring focal epilepsy in 3 canine dogs. The data from their study are freely accessible on the iEEG portal (https://www.ieeg.org/).

Table 2.1 Web-based seizure-prediction databases

Database	**Number of subjects**	**Type of EEG recordings**	**Total recordings hours (h)**	**Number of seizure**
Flint Hills	10	Intracranial	1,419	59
CHB-MIT	23	Scalp	940	198
Freiburg	21	Intacranial	708	88
Epilepsy Ecosystem	3	Intracranial	31,824	1139
European Epilepsy Database	250	Intracranical and/or scalp	>40,000	2400

2.1.2 Data preprocessing

2.1.2.1 Noise filtering

It is important to prepare the collected signal before processing and training steps. The recorded signals need to be filtered from biological and environmental noises. Additionally, recorded data may have gaps, where the devices stopped recording for a few moments, which need to be cut-off as well. Temporal filtering with digital filters has been both deployed for seizure prediction [28] [30]. Both Infinite impulse response (IIR) and Finite impulse response (FIR) filter have been considered in seizure prediction studies. Park et al. [38] studied the impact of IIR and FIR methods, and concluded that they, both, increased the sensitivity and specificity of seizure prediction.

2.1.2.2 Preictal time choice

The Melbourne University AES/MathWorks/NIH Seizure Prediction challenge [1], and The American Epilepsy Society's seizure prediction challenge[2] adopted a preictal time of 1 h prior to seizures with a fixed intervention time of 5 min. However, other studies have investigated different preictal time choices ranging from 2 min to 90 min [39] [40] [37]. In a comparative study, Teixeira et al. [33] tested, on a 278 patients, 4 different preictal times (10, 20, 30 and 40 min). His findings did not show any significant difference in sensitivity. However, the longer the preictal time, the lower the False Predicting Rate. They concluded that 30 min is the most optimum average value. Similarly, Bandarabadi et al. [34] tested the same preictal times on 24 patients and found the same optimum average value of preictal time. In a recent study on the proper selection of preictal time, Bandarabadi et al. [41] found that prediction performance is not consistent with the preictal period. Optimal duration varies from patient to patient and also changes significantly between each seizure in the same patient. The best values ranged from 5 to 173 min (44 min average).

2.1.2.3 Data segmentation

EEG signal bands range from 1 Hz to 180 Hz; thus, to avoid lost of information, sampling frequency should be greater or equal 400 Hz (according to Nyquist-Shanon rule). Accordingly, a 30 min preictal time choice would be equal to at least 720000 sample multiplied by the number of channels. Hence, it is computationally unpractical to process a complete preictal phase, and it is more convenient to break it down into smaller windows with certain overlapping rates. In the literature, the adopted windows length ranged from 1 sec [4] to 10 min [42]. Several studies adopted a non overlapping 5 sec window length [33] [32] [34], while Moghim and Come [43] added the average of features over 9 sec and 180 sec to harvest long and short term pattern analysis. Larmuseau et al. [42] used LSTM neural networks to learn current features according to long term memory to avoid small windows segmentation.

2.1.3 Feature extraction and selection

Features are either based on analyses of one EEG channel called univariate features or on extracting a combined information from multiple channels which are called multivariate features. In each category there are linear and non-linear features. In their extended review paper, Mormann et al. [30] presented a long list of features used in the literature.

[1] https://www.kaggle.com/c/melbourne-university-seizure-prediction
[2] https://www.kaggle.com/c/seizure-prediction

2.1.3.1 Univariate linear feature

Linear features have been proven to be efficient in multiple studies [33] [37] [44]. Simple linear features, such as statistical measure (variance, kurtosis, and skewness), have been reported to change in the preictal phase [45]. Spectral band power has been mostly used in seizure prediction [28]. It consists of averaging the power spectrum over each EEG band: delta (0.5-4 Hz), theta (4-8 Hz), alpha (8-13 Hz), beta (13-30 Hz) and gamma (30-128 Hz). Among all bands, findings showed that the gamma band changes the most during the preictal phase. Thus, Netoff et al. [46], suggested splitting the gamma band into four sub-bands, which mitigate the information lost after averaging the power spectrum over a wide range of frequency. Mormann et al. [27] compared the performance of 30 features and showed that Hjorth parameters (HM) are the best univariate features. Several findings showed that the EEG signal decreases its irregularity in the preictal phase. Thus, few studies [27] have adopted the error of the auto-regressive model as an input feature for classifiers.

2.1.3.2 Univariate non-linear feature

The dynamical system approach in brain modeling considers the neural system as a nonlinear dynamical system. The brain is known for having different states all of which follow certain stability and homeostasis. Such stability has implicated Hopfield networks to model brain states. The underlying dynamics of a Hopfield network could be described by the Lyapunov function. Based on the latter premise, features such as the largest Lyapunov exponent, dynamic similarity index, and correlation dimension have been deployed in seizure prediction [28]. However, inconsistent results have been reported concerning the changes in the largest Lyapunov exponent. Some studies showed a decreasing in the largest Lyapunov exponent [27] during the preictal phase while others showed the opposite [47]. Mormann et al. [27] have tested univariate nonlinear features on five different patients and reported poor performance.

2.1.3.3 Bivariate feature

Univariate features measure the changes that occur in a single recording channel which reflect a particular brain area's activity independently on other brain centers. It is been reported [48] that the preictal state influence the synchronous activity between brain areas. Le Van Quyen et al. [48] investigated the efficiency of phase-locking values of pairs of EEG channels. As a result, they found, in 70% of cases, a state of synchronization observed during the preictal phase. Although other studies [49] [50] showed a decrease in synchronization during the preictal state, processing either decreasing or increasing synchronization may lead to a

significant prediction performance.

Several seizure prediction studies included bivariate features such as the measure of lag synchronization, mutual information, mean phase coherence, dynamic entrainment, wavelet synchrony , and Shanon entropy index [27] [51] [52]. In a comparative study, Mormann et al. [27] evaluated different bivariate nonlinear features and found the conditional probability index and the Shanon entropy index to have the best performance among other measures.

2.1.3.4 Feature selection

In searching for the best measures, several studies support the utility of an extended list of features [28] [30]. Implying all supported measures would be problematic in terms of feature dimension complexity and redundancy. For most nonlinear classifiers, the order of complexity is polynomial in function of the input dimension. Thus, selecting the most discriminative features and reducing their dimensional step is recommended for seizure prediction. Several selection methods has been suggested [53] [37] [54]. In this section, we will highlight the most prominent and widely adopted algorithms, minimum redundancy maximum relevance (mRMR) [55], and genetic algorithm (GA) [56]. The mRMR sort features are based on maximum relevance and minimum redundancy. The ranking follows a cost function which can be based on different metrics. Direito et al. [57] applied mRMR to reduce feature dimensions rank from 4,410 to 132. They adopted Pearson's correlation to measure redundancy and statistical F-testing for relevance. Others [53] used mutual information metric to decrease their features number from 435 to 9. GA has been widely applied as a meta-heuristic optimization technique [58]. The algorithm is inspired by natural genetic evolution. It applies segregation and mutation on different combinations then selects the fittest and re-apply the same process to reach a certain range of fitness. Applying GA for feature selection was first proposed by Ataee et al. [59]. It was used to select the best features set and window length simultaneously. It is possible to apply different fitness functions [59] [60]; however, most of the proposed functions rely on the final performance of the classifier. In a their study, Direito et al. [57] compared the performance of GA and mRMR and concluded that the best selection method is patient-specific. Assi et al. [60] have shown, however, that a hybrid method combining GA and mRMR outperformed mRMR and GA when applied separately.

2.1.4 Classification

Seizure prediction is tackled by detecting the preictal phase which is the equivalent of learning a binary discrimination function. Several machine learning based classifiers have been studied in seizure prediction such as logistic regression, Support Vector Machine (SVM) [53], Artificial

Neural Network (ANN), Linear discriminant analysis (LDA) [61], Decision Tree, Random forest, Extreme Gradient Boost (XGB) [1], and Adaptive Neuro-Fuzzy Inference System (ANFIS) [60]. In general, in the EEG feature's space, linear classifiers can poorly discriminate the preictal phase. Additionally, EEG signals are severely imbalanced where the preictal data is significantly less than the interictal data. Classifiers tend to learn better classes with high number of samples and overfit classes with low number of samples [62]. Thus, an efficient classifier for seizure prediction needs to handle severely imbalanced datasets and non-linear discrimination. In 2015, new methods have emerged and led to a breakthrough in machine learning. Artificial neural networks became suitable for adding more layers ($>$ 100 layers), which grant them the capacity to learn deep non linear features from raw data input. The field became known as Deep Learning [63]. Additionally, a new version of gradient boosting classifier called eXtreme Gradient Boosting (XGBoost) became famous for winning Kaggle competitions [64].

In this section, we will limit our discussion to the most prominent classifiers (SVM, XGB, and ANN).

2.1.4.1 Support Vector Machine

Until recently, SVMs have been the most popularized supervised machine learning approach. SVMs are widely applied in seizure prediction [38]. [32] [60] [53] [28]. An SVM classifier [65] is trained to find the discriminative hyperplane with the maximum distance to the nearest training points. Such a technique makes SVMs highly immune to overfitting in the case of a small dataset. For a non-linear classification, SVMs can apply kernel tricks to compute non-linear decision boundaries. Several kernel functions have been explored; however, the Radial Basis Function kernel is the most applied in seizure prediction [28]. To overcome the problem of an imbalanced dataset, several studies suggested training the model on a balanced dataset [33] [34] while Park et al. [38] applied an adjusted variant of SVM called cost-sensitive SVM. This variation consists of adding more classification error penalty on classes with a lower number of samples (preictal class). SVM has been considered so far the best and most robust classifier for seizure prediction [33].

2.1.4.2 Extreme gradient boosting

XGBoost is a powerful and fast non-linear classifier that learns an ensemble of weak prediction models to build up a strong and general prediction model. The model is built gradually (stage-wise fashion) and optimized by an arbitrary loss function. The loss function should be differentiable since it is optimized through a gradient descent. XGBoost has been known for

winning most of the recent kaggle competitions where the the problem is a supervised machine learning and the input data is a set of features. The last kaggle competition hosted by the Melbourne University [1] revealed the superiority of XGB over other classification methods. Nevertheless, the number of studies applying XGB for seizure prediction is relatively low. In a recent study, Samie et al. [66] studied the possibility of predicting seizures using IoT device with constrained memory. They tested an ensemble of XGBoost and logistic regression on the epilepsy ecosystem dataset. With a simple features set (band power), they showed that XGB can perform as good as the state of the art [1].

2.1.4.3 Artificial Neural Networks

ANNs are inspired by biological neural networks. An ANN is a network that implies a set of connected units called neural networks units. In its simple version, ANNs are based on 1 or multiple layers. Each layer consists of a set of units and each unit is connected to all the units of the previous layer. Each unit output is equal to the weighted sum of all units output from the previous layer. The output is mainly fed to a non-linear function called the activation function. With multiple layers, an ANN is capable of learning a nonlinear decision boundary. Multi-layer Perceptron (MLP) is considered the simplest and the oldest version of multi-layer neural network [67]. Until the rise of deep learning and the use of graphics card for computation [63], ANNs were not considered practical to solve real machine learning problems. Only MLP with a few layers were adopted for seizure prediction [28]. Nevertheless, comparative studies [33] [68] have shown that the ANNs performance is modest compared to other non-linear classifiers.

Recently, several adjustments were introduced to ANNs architecture [69] allowing it to learn with long-term dependency and avoid over-fitting. Deep ANNs exhibit a significant performance with raw data input, due to its ability to learn in-depth features representation [70]. Additionally, deep ANNs are highly flexible and one can adjust the architecture to control the learning process. For instance, creating recursive connections allows for learning a sequential model and making skip connections enable long term gradient flow. The wide diversity of deep ANNs is now studied under the realm of Deep Learning. In the last few years, several studies proposed a Deep Learning based approach for seizure prediction [71] [72]. Korshunova [73] adopted a deep convolutional neural network (CNN) [74] which is a specific ANNs architecture highly efficient for image like input. To create an image like input, Korshunova applied a binned spectrogram on 10 min EEG recording windows. The number of the bin was equal to the number of brain signal bands and the spectrogram time window is 1 min. As a result, the input shape is equal to 10 multiplied by the number of channels and

the number of brain signal bands. Despite the severe imbalance in the dataset [75], CNNs showed a very competitive performance in seizure prediction.

2.1.5 Regularization

Despite being optimized, the classification output is prone to general false alarms. Thus, a regularization step is needed to constrain alarm generation. Methods considered in the literature are temporal adaptive filters such as Kalman filter (KF) or firing power (FP) technique [28].

FP is a simple quantification technique that counts the number of predictions classified as preictal within a sliding window. An alarm is generated only when the number of firing exceeds a certain threshold. Multiple studies have applied FP technique in their seizure prediction algorithms [34] [33] [76]. Mainly they adopted a fixed normalized threshold (0.5) [34] [33]. Teixeira et al. [76] tested twenty thresholds values (0.1, 0.15, 0.2,. . . , 0.85), and concluded that threshold is more specific to each patient; however, low threshold values are generally more conservative in raising alarms.

KF is an adaptive filter that learns a linear quadratic estimation [77]. It is applied on a series of measurements over time, which is believed to be noisy or inaccurate. In seizure prediction, Chisci et al. [78] were the first who used KF in seizure prediction. They compared the performance of the classifier with KF regularization and without and showed a significant improvement when applying regularization. Park et al. [38] adopted KF, as well, and showed optimistic results.

In their research for the best regularization method, Teixeira et al. [76] compared KF and FP and found the latter to be more specific in raising alarms. However, KF showed to be more accurate in terms of sensitivity. Texeiria et al. considered the FP technique to have a longer memory that is proportional to the window length. Hence, it tends to be more conservative than one or two step adaptive filters.

Our group [79] trained deep recurrent ANNs to learn the best regularization function. We showed that ANNs can combine a long memory (up to 100 time steps) and a highly adaptive function (completely learned from data). Our findings confirmed Teixeira et al. [76] comparasion between KF and FP method and proved that with decent dataset, ANNs-based method are more conservative than FP technique.

2.1.6 Discussion

Seizure prediction has made significant progress in the last few years. The EEG databases became more optimized for seizure prediction studies [1]. Additionally, the availability of big datasets for the public and launching public competitions [4] [75] brought people from outside the field to contribute. Moreover, public competitions allowed crowd-sourcing hundreds of different methods and approaches which are equivalent to dozens of comparative studies. The astonishing results obtained in the last Kaggle competition were based, mostly, on an ensemble of kernels and classifiers [80]. However, models with a complex structure can be unpractical for embedded and portable devices. Thus, it is essential to study how to reduce the model complexity with minimum impact on performance [66].

Despite the remarkable breakthrough of deep learning in the last few years, studies investigating the utility of deep learning in seizure prediction are still humble. In most cases, deep learning is only considered as a discriminator. Nevertheless, the ability of deep ANNs to learn deep features and discriminate only using raw data has not been exploited yet. Deep learning is also productive for dimension reduction of the feature set. It is faster and more customized than GA. Additionally, deep learning has shown some progress in meta-learning which teaches the model to generalize its knowledge to unseen datasets with minimum samples. Such a paradigm can be a transition from patient-specific seizure prediction to bench-marked models.

CHAPTER 3 THEORY AND METHODOLOGY

3.1 Data preprocessing

EEG signals are prone to noises and artifacts. As explained in Chapter 1, artifacts can be a result of biological factors (e.g., patient movement, muscle contraction) or environmental factors (e.g., the recording machine, the electrodes). To preserve the information in EEG rhythmic activities (5 EEG bands) we applied a bandpass filter with cutoff frequencies equal to 2 Hz and 180 Hz. The digital filter can be either a Finite Impulse Response filter (FIR) or an Infinite Impulse Response filter (IIR). The term 'Impulse Response' means the response of the filter to an impulse in the time domain. In the case of IIR, the output of the filter depends on the input and the output of the previous time step(s).

$$y(n) = \sum_{k=0}^{N} a(k)x(n-k) + \sum_{j=0}^{P} b(j)y(n-j) \tag{3.1}$$

The FIR, on the other hand, produces an output independent of the prior outputs.

$$y(n) = \sum_{k=0}^{N} a(k)x(n-k) \tag{3.2}$$

In Eq. 3.1 and 3.2, $y(n)$ refers to the output time history, $x(n)$ refers to the input time history, and $a(k)$ and $b(k)$ are terms from the filter transfer function. Practically, an IIR filter has a relatively lower order, hence, fewer computations are required to achieve the same results with FIR filters, which makes the IIR filter computationally faster. However, an IIR filter suffers from instability and has a nonlinear phase response. FIR filter is slow and steady. In both filters, the order of the filter is equivalent to the number of terms in the equation. The higher the order is, the sharper the cut-off edge becomes. However, with similar order, IIR filters are significantly sharper than FIR filters. It will require the 40th order of an FIR filter to obtain the same sharpness as the 10th order of an IIR filter. The time delay is also a function of the order of the filter. A filter with high order has a high time delay. FIR filters have a linear time delay independent of the input frequencies, while IIR filters time delay is non-linear and depends on the frequencies. A time delay can be solved by applying zero phase filtering. It consists of reapplying the same filter on the output sequence but in the opposite direction. Zero-phase filtering comes with a trade-off where the calculation takes twice as long to be performed.

Table 3.1 The main characteristics of different FIR filters

Method	Pass band	Transition width	Stop band
Chebyshev	Ripple	Narrow	Monotonic
Bessel	Slopping	Very wide	Slopping
Butterworth	Flat	Wide	Monotonic

In our study, the main concern is not related to the speed of prediction. Therefore, we adopted FIR filters for the preprocessing step. There are several types of FIR filters and to choose, one should consider the specifications that depend on many variables. Each filter method has its characterizations in pass-band, transition width, and stop-band. Table 3.1 displays the characteristics of the most prominent FIR filters. We applied a Chebyshev passband filter because of its narrow transition and monotonic stopband.

In most cases, electrodes recording the EEG signals can be different in terms of sensitivity and amplitude scale. Thus, exposing the classifier to an unbalanced voltage amplitude can impose a significant bias. To solve this, we normalized the amplitude of each channel signal by dividing each value by the maximum value. Next, we normalized it as described in Eq. 3.52 to obtain a zero centered distribution.

$$X_i = \frac{x_i - x_{mean}}{\sigma} \tag{3.3}$$

3.2 Features extraction

As mentioned in Chapter 2, the general performance of a classifier highly depends on the features set. It has been shown that univariate linear features have superior predictive results. In our research, we aim to test different regularization functions. Therefore, we adopted a simple feature set with high predictive information, which consisted of statistical measures (variance, skewness, kurtosis) and EEG's bands power (delta, theta, alpha, beta, and gamma).

3.2.1 Statistical measures

Considering a discrete measure sample x_i, the variance reflects the sparsity of any distribution and it is the average of the squared distance of each sample and the mean (Eq. 3.4).

$$Var = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu)^2 \tag{3.4}$$

The asymmetry in a statistical distribution is called skewness. It describes the extent to which a distribution is unlike the normal distribution (Eq. 3.5).

$$\gamma_1 = \mathrm{E}\left[\left(\frac{X-\mu}{\sum}\right)^3\right] \tag{3.5}$$

The kurtosis is the 4th statistical moment. It reflects the tailedness of the distribution, which is equivalent to the study of the flatness of the amplitude distribution (Eq. 3.6). Seizure prediction has made significant progress in the last few years. The EEG databases became more optimized for seizure prediction studies [1]. Additionally, the availability of big datasets for the public and launching public competitions [4] [75] brought people from outside the field to contribute. Moreover, public competitions allowed crowd-sourcing hundreds of different methods and approaches, which is equivalent to dozens of comparative studies. The astonishing results obtained in the last Kaggle competition were based, mostly, on an ensemble of kernels and classifiers [80]. However, models with a complex structure can be unpractical for embedded and portable devices. Thus, it is essential to study how to reduce the model complexity with minimum impact on performance [66].

Despite the remarkable breakthrough of Deep Learning in the last few years, studies investigating the utility of Deep Learning in seizure prediction are still humble. In most cases, Deep Learning is only considered as a discriminator. Nevertheless, the ability of deep ANNs to learn deep features and discriminate only using raw data has not been exploited, yet. Deep Learning is also productive for dimension reduction of the feature set. It is faster and more customized than GA. Additionally, Deep Learning has shown some progress in meta-learning, which teaches the model to generalize its knowledge to unseen datasets with minimum samples. Such a paradigm can be a transition from patient-specific seizure prediction to benchmarked models.

$$Kurt[X] = \mathrm{E}\left[\left(\frac{X-\mu}{\sum}\right)^4\right] \tag{3.6}$$

3.2.2 Relative spectral band power

According to Fourier analysis, any time domain signal can be broken down into many discrete frequencies or a spectrum of frequencies over a continuous range. A periodogram or Welch's estimation of spectral density estimation is the Fourier transform of the auto-correlation of a time series signal(Eq. 3.7.

$$\mathcal{F}\{x(t) * x(-t)\} = X(f) \cdot X^*(f) = |X(f)|^2 \tag{3.7}$$

The absolute spectral band power is the area under the curve of power spectrum bounded by the range of a given band (between 4 and 7 Hz in theta band). By dividing the absolute spectral band power by the total spectral power (total area under the curve), we obtain the relative spectral band power which describes the percentage of the total power of the signal. EEG signals are known to contain significantly more power in the range of low frequencies. Therefore, to avoid biases in the classifier training, we applied a feature scaling function to our features set (Eq. 3.8).

$$X_i = \frac{x_i - x_{min}}{x_{max}} \tag{3.8}$$

3.3 Machine Learning classifier

Machine Learning (ML) is realm in Artificial Intelligence based on teaching computers to learn how to solve problems based on human or engine experiences or so called data samples. Machine Learning can be applied to many problems. Any field that needs to interpret and act on data can benefit from Machine Learning techniques. Basically, Machine Learning aims to learn and optimize the prediction function with respect to the objective function. It applies many methods to update the function in order of maximizing the objective score. Formally, a Machine Learning system adopts a statistical learning approach, learn to model the right probability distributions and uses them for inference and future prediction. Machine Learning has the advantage to estimate an accurate prediction solely by learning from data without any prior knowledge of the real model dynamics. Additionally, it helps in approximating solutions for intractable objective function.

In his review article [81], Mitchell provides a concise definition:"A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E." For example, a Machine Learning model that learns to play chess might increase its performance and its ability to win at the class of tasks including playing chess, through experience collected by playing chess against itself. In general, a Machine Learning problem is defined by the following features: the the category of tasks or the objective function, the performance metrics, and the dataset.

In conclusion, Machine Learning is reliable for: (a) problems that have been solved with a hand engineered parameter tuning, (b) complex problem with no-existing solution which falls

beyond human modeling state of the art, hence a better solution is optimized and learned from training data can capture (c) non stationary environment where heuristic optimal solutions change with time, and (d) obtaining insights about complex problems and big data. There are different categories of Machine Learning algorithms: supervised learning, unsupervised learning, and reinforced learning. In this study, we are essentially concerned with supervised learning.

3.3.1 Supervised Learning

In supervised learning, the training data one use for training is associated with the correct solutions, called labels. A supervised learning algorithm takes labeled data and train a model to predict according to the data labels. Supervised learning problems can be either a classification problem or a regression problem. In a classification problem, the labels are categoraly and the model is challenged to classify the input to the right category. In a regression problem, there is a relationship to be determined among many different variables. Usually, this takes place in the form of prior data being used to predict future samples. An example of this would be predicting the future weather features according the weather history.

There are many supervised learning methods, however, in this study we consider the most popular algorithms: 1) logistic regression for regression problems 2)support vector machines for classification problems and 3) Artificial Neural Network.

3.3.1.1 Logistic Regression

Although called regression, Logistic Regression is a classification method which belongs to probabilistic classifiers. It is based on predicting the probability of a class conditioned to the input vector. Logistic regression starts with applying a linear combination of the input dimensions. The predicted value is an unbounded real number. The aforementioned value is passed to a sigmoid function that maps the boundless value to a decimal number that ranges from 0 to 1. The sigmoid function (Eq. **??**) output is considered the probability of the predicted value.

$$\frac{1}{1 + \exp -k(x - x_0)} \tag{3.9}$$

Logistic regression project input values using a linear combination with weights or coefficient values to predict an output value (Eq. 3.10).

$$y = \frac{\exp b + wx}{1 + \exp b + wx} \tag{3.10}$$

where b is the bias or offset, w is the coefficient for the single input value (x) and y is the prediction output. Input dataset can be reprsented by a matrix X which has N comlumn $\boldsymbol{x}$ where N represents the number of data samples. Each column in the dataset is associated with a b coefficient (a constant bias) which is optimized during the training phase. Logistic regression predict the probability of the default class and model the probability that the class Y of an input x is equal to 1, which can be formulated as:

$$P(X) = P(Y = 1|X) \tag{3.11}$$

The predicton function in Logistic Regression is a linear combination; however, the predictions are mapped using the logistic function. Thus, the predictions are no longer considered as a linear combination of the inputs as with the linear regression, for example, continuing on from above, the model can be stated as:

$$P(x) = \frac{\exp b + wx}{1 + \exp b + wx} \tag{3.12}$$

The coefficients of model must are learned from the training data by using maximum-likelihood estimation. Maximum-likelihood estimation is a common learning algorithm widely applied in Machine Learning algorithms. It assumes a prior distribution of data (mainly normal) and its objective is equivalent to minimizing the negative log likelihood of the prediction function.

$$\begin{aligned} L(\theta \mid x) &= \Pr(Y \mid X; \theta) \\ &= \prod_i \Pr(y_i \mid x_i; \theta) \\ &= \prod_i h_\theta(x_i)^{y_i}(1 - h_\theta(x_i))^{(1-y_i)} \end{aligned} \tag{3.13}$$

The optimum model coefficients would generate a prediction value very close to 1 for the positive class and a value very close to 0 for the other class. The motive of maximum-likelihood for logistic regression is to optimize the coefficients vector with respect to the error in the probabilities predicted by the model to those in the data (e.g. probability of 1 if the data is the primary class).

3.3.1.2 Support Vector Machine

SVM is one of the most popular and efficient machine learning algorithms. They are used for both classification and regression and were extremely popular around the time they were developed in the 1990s and stills considered the best solution for different problems where the input features are categorical and with a limited training set. SVMs are trained by finding a linear hyperplane that separates the training dataset into two classes. SVMs are able to avoid local optimal solution by find the optimal separating hyperplane which is intuitively computed when the margin, to the nearest training data samples is as large as possible. The hyperplane that has the maximum separation distance has the least chance of overfitting and the best ability for generalization. Mathematically, SVM is a maximum margin linear model. Given a training dataset of n samples of the form $\{x1, y1, \ldots, xn, yn\}$ where x_i is an n-dimensional feature vector and $y_i = \{1, -1\}$ is the class to which the sample xi belongs to. The goal of SVM is to find the maximum-margin hyperplane which divides the group of samples for which $y_i = 1$ from the group of samples for which $y_i = -1$ .This hyperplane can be presented as the set of sample points satisfying the following equation:

$$\mathbf{w}^T\mathbf{x}_i + \mathbf{b} = 0 \tag{3.14}$$

where w is the normal vector to the hyperplane and any samples above the hyper the plane should have label 1, i.e., x_i such that

$$\mathbf{w}^T\mathbf{x}_i + \mathbf{b} > 0 \tag{3.15}$$

will have corresponding $y_i = 1$. Similarly, all samples under the hyperplane are labels 1, i.e., x_i such that

$$\mathbf{w}^T\mathbf{x}_i + \mathbf{b} < 0 \tag{3.16}$$

will have corresponding $y_i = -1$. The sample data is rescaled such that anything on or above the hyperplane

$$\mathbf{w}^T\mathbf{x}_i + \mathbf{b} = 1 \tag{3.17}$$

is of one class with label 1, and anything on or below the hyperplane

$$\mathbf{w}^T\mathbf{x}_i + \mathbf{b} = -1 \tag{3.18}$$

is of the other class with label -1. To predict an unknown point, it is enough to predict

whether it belongs to a positive class or negative class. If $\mathbf{w}^T\mathbf{u} + \mathbf{b} > 0$ (where u is an unknown point, and w is the final vector) then it belongs to the class with label 1, else it belongs to the class with a label -1.

SVM differs from other Machine Learning classifiers by adding condition on the objective function. The algorithm does not limit itself to train a model with minimum error, but rather finding the hyperplane that discriminates two classes by maintaining the highest distance between them. SVM is generalized by applying nonlinear kernels to learn a nonlinear manifold that separates two entity that is linearly inseparable. The optimum hyperplane in linear SVM can be computed directly using matrix operations. A simple intuition is that the linear SVM can be reformulated based on the inner product of two given observations, rather than the observations themselves. The inner product of two vectors is equal to the sum of the multiplication of pair wise multiplication. The equation for making a prediction for a new input using the dot product between the input ($\mathbf{x}_i$) and each support vector ($\mathbf{x}_i$) is calculated as follows:

$$f(x) = b + \sum_{i=1}^{n} a_i x_i x \tag{3.19}$$

This is an equation that involves calculating the inner products of a new input vector (x) with all support vectors in training data. The coefficients B0 and ai (for each input) must be estimated from the training data by the learning algorithm.

Despite the advantages of SVM, there are a few limitations. First, when choosing a Gaussian kernel, it is hard to find the best parameters and trying multiple values is hard because of the second reason. The second limitation is the speed and size, both in training and testing. It requires highly complex and time-consuming calculations. This is considered, the most serious problem with SVMs, and it is described as a high complexity algorithm with extensive memory requirements. In large scale dataset, it requires computations with quadratic complexity. Many machine learning problems become exceedingly difficult when the number of dimensions in the data is high. This phenomenon is known as the curse of dimensionality. Of particular concern is that the number of possible distinct configurations of a set of variables increases exponentially as the number of variables increases.

3.3.2 Deep Learning

Deep Learning is a realm of machine learning based on ANNs. It leverages the potential ANNs in learning highly nonlinear function and exploit their ability to linearly scaling their training time. Deep Learning is the expression of stacking multiple (can exceed hundreds) layers of

ANNs and trains it all end to end as a monolithic model. This process seems to provide a deep and hierarchical learning representation of the input domain why is called Deep Learning. In the last five years, thanks to the explosion of data and the revolution in parallel computation, Deep Learning succeeded in bringing a new era of AI where it dramatically improved state of the art in computer vision, speech recognition, natural language processing, generative models, etc. Traditional Machine Learning methods suffer from the curse of dimensionality, where one should consider extracting hand-engineered features to reduce the input dimension, which creates another problem of choosing the best feature set. However, Deep Learning has shown to be highly efficient in processing raw data, especially in the pixel domain. Deep Learning prevailed over other Machine Learning methods because of the following properties

- Simplicity: Deep Learning methods can learn feature representation with minimum preprocessing. Thus, it skips the feature extraction and selection step, which is often handcrafted.
- Scalability: Deep learning models are linearly scalable to big datasets. Other competing methods (e.g., kernel machines) requires computational time that grows exponentially with dataset size.
- Domain transfer: A model learned on one task applies to other related tasks, and the learned features are general enough to work on a variety of tasks which may have small data available.

3.3.2.1 Deep Feedforward Artificial Neural Networks

ANNs are considered the building block of Deep Learning models. They manifest a parallel structure and adaptive capabilities to train on extensive scale data, which is regarded as the key behind the breakthrough in different machine learning applications. The basic building block of a neural network is a "neuron unit," which can be perceived as a processing unit. In a neural network, each neuron is connected to all neurons of the previous layer where each connection is parametrized by a weight coefficient of w. The neuron output is the weighted combination of the all connection input transformed by a nonlinear activation function (sigmoid, rectified linear, tangent). Multiple neurons can form a layer which itself can be an input for the next layer. The aforementioned basic structure is called Multi-Layer Perceptron (MLP) and is considered the most primitive form of ANNs. MLPs or Feed Forward ANNs are very rich in parameters which requires a very large size of the dataset to avoid overfitting. Additionally, their capacity to maintain efficient training with a high number of layers is limited due to the phenomenon of gradient vanishing and explosion in long term

dependency. Many architectures were proposed aiming to provide an inductive bias for the ANNs structure related to the problem definition, which hinders the sensitivity of MLPs toward input noises. For sequential input where the input is a sequence in time, Recurrent Neural Networks are considered the best choice. While for images, a neural network built from the convolutional mask is state of the art.

ANNs are trained by the backpropagation algorithm, which is considered the bedrock of the Deep Learning optimization and the solution for building multiple layers that are trained end to end. The backpropagation algorithm is a nutshell a derivation of the loss function with respect to each weight matrix in the network following the chain rule for partial derivative.

3.3.2.2 Convolutional Networks

Convolutional Networks [74], also known as convolutional neural networks (CNNs) are a specific type of ANNs that are designed to process input samples organized in a grid-like shape. It had an enormous impact on the state of the art of computer vision. Additionally, it proves efficient in processing one-dimensional sequential arrays (Wavenets). CNNs are based on the convolutional operation, which is computed by passing a mask with a unique center. The mask, or kernel, is a linear function that projects the weighted multiplication of all inputs under the mask to the input under the center of the mask. Next, the convolutional

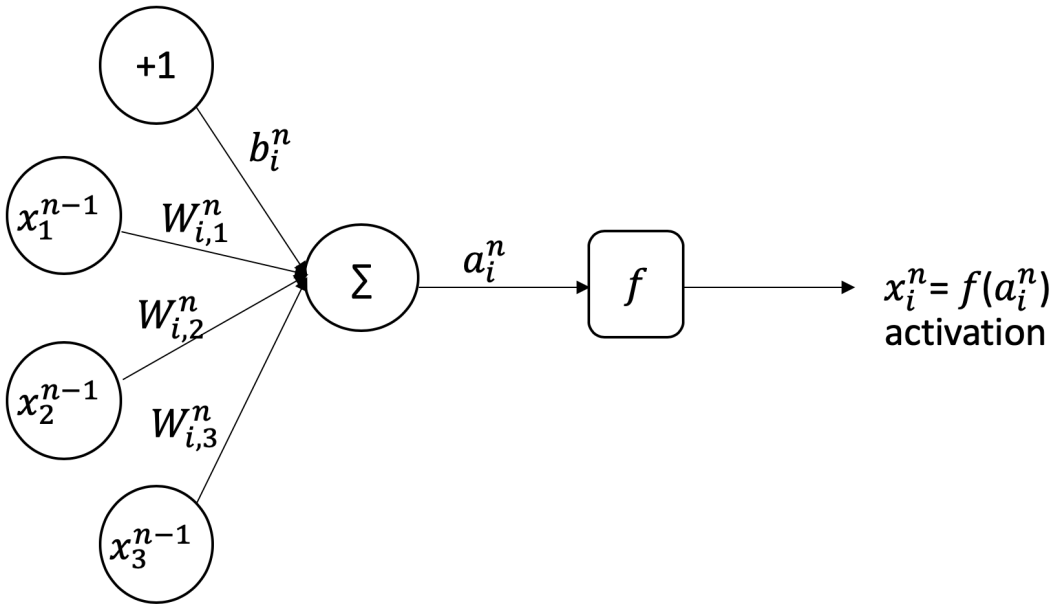

Figure 3.1 Simple activation neural unit. It applies weighted sum on the input before applying a non-linear activation functionn

mask strides one step and repeats the operation until it passes on the whole input grid.

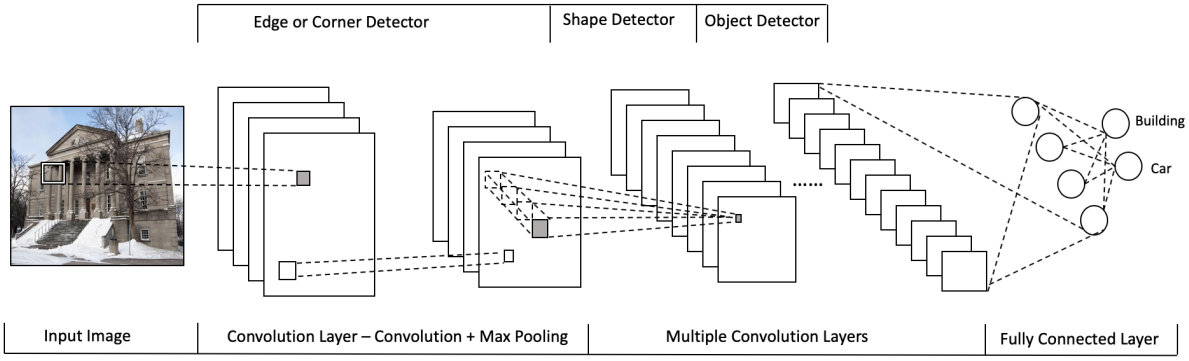

Figure 3.2 Simple architecture of convolutional network. It can range from 1 layer to 120 layers

For example, in the case of a two-dimensional input, the equation of a discrete convolutional mask is the following:

$$S(i,j) = (I * K)(i) = \sum_m \sum_n I(m,n)K(i-m, j-n) \tag{3.20}$$

where I is the input, K is the kernel, m and n are the dimensions of the image. $S(i,j)$ is the output of the kernel operation on pixel (i,j).

CNNs start with a layer of multiple kernel functions. Kernel's parameters are trained by backpropagation, and next, the total images filtered by these kernels are called feature maps. To reduce the dimension of feature maps, a pooling step is required. Pooling consists of dividing the feature map into squares and applying a maximum or average function to each square. Thus, the dimension is reduced which makes it possible to go for multiple layers and the semantic information are preserved. After adding multiple layers, a flattening step of all features map is applied before feeding it to a feedforward neural network and applying multiple class logistic regression (Figure 3.2).

In case of post-processing regularization, a variation of CNN called 1D CNN (Fig 3.3) can be applied on the one dimensional probability sequence. 1D CNN is a CNN that follows the exact method except that we apply a one dimensional instead of a two dimensional convolutional kernel. The convolution equation becomes as follows:

$$S(i) = (I * K)(i, j) = \sum_m I(m)K(i - m) \tag{3.21}$$

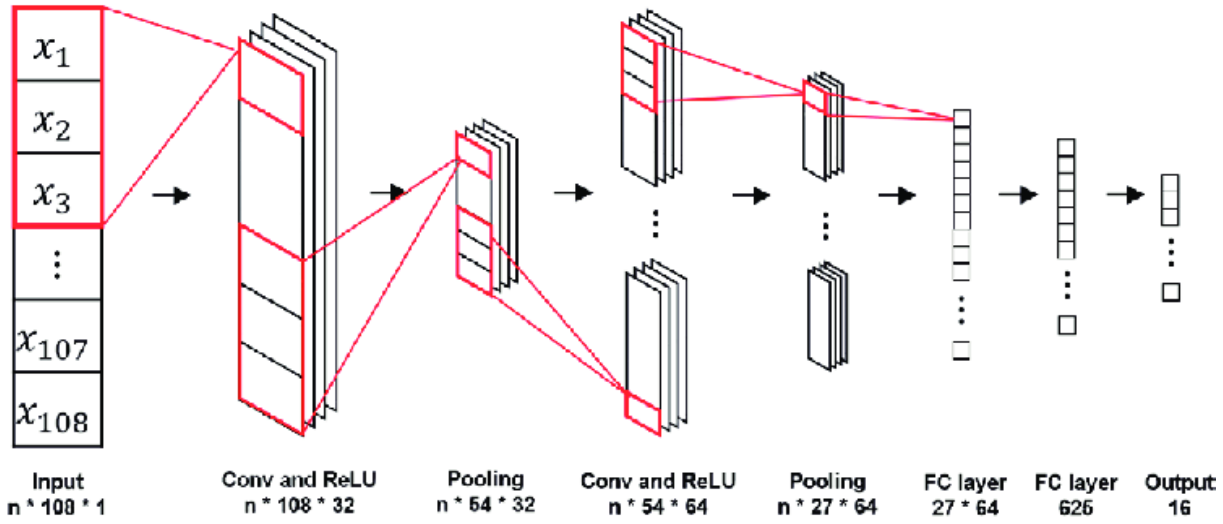

Figure 3.3 Simple architecture of one dimentional convolutional network [5]

3.3.2.3 Recurrent Neural Network

Sequential data have a special property where a sample X_i drawn at time i is dependent on $X_{i-1}....X_{i-k}$. Such a property requires that the ANNs be able to learn the features of the current input X_i considering the features of the previous input. In a fully connected feed forward neural network, parameters are optimized and trained depending on the input

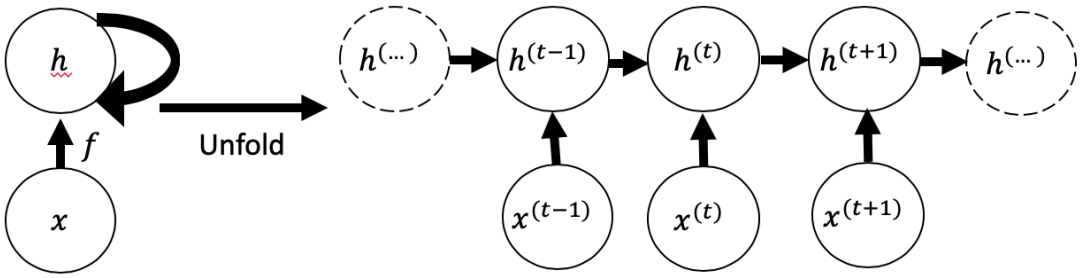

Figure 3.4 Simple architecture of recurrent neural network.

features of a specific position. For instance, in the sentence "I was born in 1989", if a feedforward network is trained on this sentence it will learn about the year in the last neural units since the year occurred last in the phrase. If this sentence tests the same network: "In 1989, I was born", the neural units assigned to the year will not be activated. Fortunately, one can apply recurrent neural networks (RNNs) to build a sequential-based knowledge of the dataset. An RNN (Figure 3.4) takes a sequence of vectors $\mathbf{x}^t$, where t is the time step of a sequence with length τ. At each time step, a function $\mathbf{h_t}$ multiplies $\mathbf{x}_t$ with input weight matrix $\mathbf{U}$ and multiplies $\mathbf{h_{t-1}}$ with hidden layer to hidden layer weight matrix $\mathbf{W}$. Next, $\mathbf{h_t}$ adds the two entities and applies a non-linear activation function. Depending on the RNNs architecture, the output $o^{(t)}$ is the linear combination of weight matrix $\mathbf{V}$ and $\mathbf{h_t}$. $\hat{y}^{(t)}$ is the softmax prediction output at time t.

$$
\begin{aligned}
\boldsymbol{a}^{(t)} &= \boldsymbol{b} + \boldsymbol{W}\boldsymbol{h}^{(t-1)} + \boldsymbol{U}\boldsymbol{x}^{(t)} \\
\boldsymbol{h}^{(t)} &= \tanh\left(\boldsymbol{a}^{(t)}\right) \\
\boldsymbol{o}^{(t)} &= \boldsymbol{c} + \boldsymbol{V}\boldsymbol{h}^{(t)} \\
\hat{\boldsymbol{y}}^{(t)} &= \text{softmax}\left(\boldsymbol{o}^{(t)}\right)
\end{aligned}
\tag{3.22}
$$

Despite the great performance in sequential data, RNNs suffer from a serious limitation. When the input time length is long (more than 50-time steps), the network fails to propagate the gradient steadily. The gradient of the loss function with respect to the weight of the first layers is, according to the chain rule, the product of the gradients of the intermediate layers. In an analytical study, Pascanu et al. [82] provided a mathematical proof showing that if the gradient matrices are not orthogonal, the gradient will inevitably vanish or explode. Different solution has been proposed to solve the long-term dependency problem in RNNs [82] [83] [84]. The most efficient alternative was a variation of RNNs proposed by Gers et al. [84], and is called Long Short Term Memory network (LSTM). LSTM is a recurrent neural network that is composed of a memory block. Each block embeds different gates (networks) that learn what to forget from previous time features, what to memorize, and what to produce as an output.

LSTM has three vectors: input vector $\mathbf{x}_t$, hidden output vector $\mathbf{h}_{t-1}$, and the state vector $\mathbf{c}_t$. These vectors are used with the weights matrices to compute the gate functions. There are three main gates, the input gate, the output gate, and the forget gate. Each gate plays a role in producing the state vector which, by turn, controls the hidden $\mathbf{h}_t$ (Eq. 3.23).

$$
\begin{aligned}
f_t &= \sigma_g\left(W_f x_t + U_f h_{t-1} + b_f\right) \\
i_t &= \sigma_g\left(W_i x_t + U_i h_{t-1} + b_i\right) \\
o_t &= \sigma_g\left(W_o x_t + U_o h_{t-1} + b_o\right) \\
c_t &= f_t \circ c_{t-1} + i_t \circ \sigma_c\left(W_c x_t + U_c h_{t-1} + b_c\right) \\
h_t &= o_t \circ \sigma_h\left(c_t\right)
\end{aligned}
\tag{3.23}
$$

W matrices, **U** and b are the weights and the biases that need to be learned during the training. Figure (3.5) illustrates the inference of different gates in an ordinary LSTM unit.

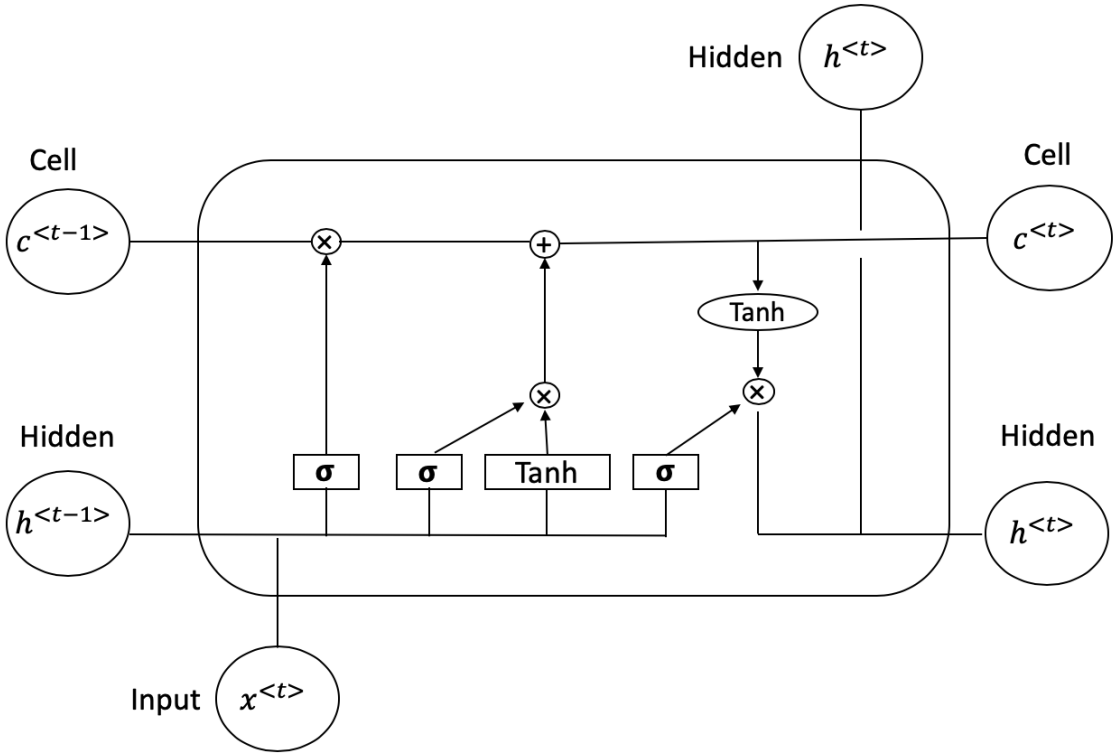

Figure 3.5 The general pipeline of LSTM architecture.

3.4 Postprocessing regularization

3.4.1 KF

KF [77] is an algorithm that recursively takes as input a sequence of measures, believed to be inaccurate or infected by noises, and projects an estimation of the future values of the input. Its estimation tends to be more accurate and adaptive to the change of the input's pattern. KF has a wide application in technology. It has been deployed for robot mapping and control, for navigation systems, and for signal processing of sequential data. KFs are based on 3 major blocks: Measure, Update, and Predict.

3.4.1.1 Preliminaries

Any KF implies a state variable x. x can be a scalar or a vector and it represents the features we want to estimate and predict. Next, a model is needed to be defined. The model represents a function that describes the system behavior. In ordinary KF, the model is a linear function of the state (Eq. 3.24). For example, in the case of a falling object, knowing the position and the speed of the object allows us to estimate its next position based on the model dynamics.

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{F}_k \hat{\mathbf{x}}_{k-1|k-1} + \mathbf{B}_k \mathbf{u}_k + \mathbf{W}_k \tag{3.24}$$

where $\hat{\mathbf{x}}_{k|k-1}$ is our model based prediction for the state in the time k considering $k-1$; $\mathbf{F}_k$ is the matrix expression of the model; $\mathbf{B}_k$ is the control-input model which is applied to the control vector $\mathbf{u}_k$; and $\mathbf{W}_k$ is the process noise with a zero mean multivariate normal distribution, $\mathcal{N}$, with covariance, $\mathbf{Q}_k$: $\mathbf{w}_k \sim \mathcal{N}(0, \mathbf{Q}_k) \mathbf{w}_k \sim \mathcal{N}(0, \mathbf{Q}_k)$.

In our study, a passive model will be considered and for simplicity, we will ignore the two variables $\mathbf{B}$ and $\mathbf{u}$.

The predicted vector $\hat{\mathbf{x}}$ is compared to the observed measures $\mathbf{x}$. The covariance of the error is the matrix $\mathbf{P}$, and the it is update along with the state by the following:

$$\hat{\mathbf{P}}_{k|k-1} = \mathbf{F}_k \hat{\mathbf{P}}_{k-1|k-1} + \mathbf{Q}_k \tag{3.25}$$

Next, the updated block takes care of updating the optimal Kalman gain, $\mathbf{K}_k$,

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^{\mathsf{T}} \mathbf{S}_k^{-1} \tag{3.26}$$

where $\mathbf{H}_k$ is the observation model which maps the true state space into the observed space. It is equal to the identy matrix in case where the predicted states are the same as the observed ones. $\mathbf{S}$ is the covariance matrix of $\tilde{\mathbf{y}}$, where:

$$\tilde{\mathbf{y}}_k = \mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_{k|k-1} \tag{3.27}$$

$\tilde{\mathbf{y}}$ is used to update the state $\mathbf{x}$ based on the weighted difference between the previous estimation to the current state and the current observation (eq 5).

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k \tilde{\mathbf{y}}_k \tag{3.28}$$

The estimate covariance $\mathbf{P}$ is also updated:

$$\mathbf{P}_{k|k} = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)\, \mathbf{P}_{k|k-1}\, (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k)^{\mathsf{T}} + \mathbf{K}_k \mathbf{R}_k \mathbf{K}_k^{\mathsf{T}} \tag{3.29}$$

An initialization step the Initial System State ($\hat{\mathbf{x}}_{0|0}$), and the Initial State Uncertainty ($\mathbf{P}_{0|0}$) precedes the main pipeline. Estimating the noise covariances $\mathbf{Q}$ and $\mathbf{R}$ for a practical performance is remarkably difficult. A practical approach is to use the lime-lagged autocovariances of operating data to compute the covariances; such technique is called the autocovariance least-squares (ALS) [85].

3.4.1.2 KF for regularization in seizure prediction

In our experiments, we adopted the same approach proposed in [78] to obtain similar regularization performance using KF. A supervised Machine Learning model $\boldsymbol{M}$ takes features input and compute a normalized output $\boldsymbol{z}$ equivalent to the probability of having a preictal phase. $\boldsymbol{z}$ is obtained at each time step k and is assumed to be affected by noise. d_k represents the noiseless signal,

$$z_k = d_k + v_k \tag{3.30}$$

where v_k is an observation white noise with a given standard deviation $\sum_v$. To build an estimation model, we assume $\dot{d}_k$ to be the rate of changes of d_k. d_k and $\dot{d}_k$ are represented by the two dimensional vector s_k. The model is built on the premise that the ideal probability of preictal phase increases smoothly when approaching a seizure episode. $\tilde{s}_k$ is the model based estimation for the state at time k. Hence, we can model the dynamic through this equation:

$$\begin{cases} \tilde{s}_k &= \begin{bmatrix} 1 & T_p \\ 0 & 1 \end{bmatrix} \hat{s}_{k-1} + w_k \\ z_k &= \begin{bmatrix} 1 & 0 \end{bmatrix} s_k + v_k \end{cases} \tag{3.31}$$

where T_p is the prediction interval which is equal to the time step (in sec) considered to compute one prediction. w_k is the process disturbance mentioned in Eq. 3.24. Its covariance matrix Q

$$\mathbf{Q} = \begin{bmatrix} \sum_w^2 \frac{T_p^3}{3} & \sum_w^2 \frac{T_p^2}{2} \\ \sum_w^2 \frac{T_p^2}{2} & \sum_w^2 T_p \end{bmatrix} \tag{3.32}$$

where $\sum_w$ is the standard deviation of the posterior error covariance matrix P (Eq. 3.25). We adopted a Kalman optimal gain as a constant value equal to $\sum_w / \sum_v$. The estimates are corrected with respect to the Kalman gain, the estimation error, and the previous prediction.

We fixed the value of $\sum_v$ (unity value) and tuned the cost of $\sum_w$) which is equivalent to tweaking the Kalman gain. As a result, the algorithm takes a measure z_k, then compute $\tilde{s}_k$ using initialized value for time t=0. By obtaining z_k and $\tilde{s}_k$, we can make the prediction $\hat{s}_k$ which is recursively used for step 1.

$$\hat{s}_k = \tilde{s}_k + \mathbf{K}_k(z_k - \begin{bmatrix} 1 & 0 \end{bmatrix} \tilde{s}_k) \tag{3.33}$$

3.4.2 Firing power technique

The FP method produces alarms considering the temporal dynamics of the decision output. In case of classifiers with continuous outputs, the first step is to binarize the probability output according to:

$$o_k = \begin{cases} 1, & \text{if } p_k \geq 0.5 \\ 0, & \text{if } p_k < 0.5 \end{cases} \tag{3.34}$$

Next, we apply a sliding window with size τ. The goal is to quantify the number of samples classified as preictal within a time step τ to:

$$f = \frac{\sum_{k=n-\tau}^{n} o_k}{\tau} \tag{3.35}$$

where o_k is the decision output of the Machine Learning classifier, and f is the "firing power"

at the discrete time n corresponding to the number of averaged samples considered in each time segment τ. An FP of value one (a full firing power) means that all the samples in the past preictal time were classified as preictal, therefore, strongly suggesting a preictal state. The alarms are generated

$$a[n] = \begin{cases} 1, & \text{if } fp[n] \geq L \\ 0, & \text{if } fp[n] < L \end{cases} \tag{3.36}$$

based on a threshold value. The threshold can vary from generating an alarm only if all the outputs in the time window τ correspond to the preictal phase, to producing an alarm whenever a preictal output is met.

3.5 Regularization from information theory perspective

In literature [76], the regularization function is either a static function that counts the firing rate and acts based on a threshold (FP technique) or an adaptive function with a relatively short memory that smoothes the prediction based on an estimation error (KF). Before discussing the limitation of each method, we introduced a brief interpretation of the information quantity in the sequence of the classifier outputs.

3.5.1 Entropy of a vector of information

In this section, we introduce an analytical demonstration showing that using a probability or a float number output of the classifier has an extreme advantage from an information theory perspective. A Shanon entropy H is a unit that quantifies the amount of information in a variable or a bit (Eq. 3.37)

$$H = -\sum_{x} p(x) \log_2 p(x) \tag{3.37}$$

where $p(x)$ is the probability of the random variable X to be equal to x. The intuition behind this measure is based on the fact that any predictable news or information (e.g., the sun will rise each day) brings no value if we know that it will happen. On the other hand, a decidedly less likely event (e.g., the president will be assassinated tomorrow) is considered valuable and hold too much information in the same small phrase.

Given a vector $\mathbf{x}$ which represents a sequence of N random values $X_1, X_2, X_3, ..., X_N$

The entropy of $\mathbf{x}$

$$\begin{aligned} H(\mathbf{x}) &= -\sum_{\mathbf{x}} p(\mathbf{x}) \log_2 p(\mathbf{x}) \\ &= -\sum_{X_1} \sum_{X_2} ... \sum_{X_N} p(X_1, X_2, ..., X_N) \log_2 p(X_1, X_2, ..., X_N) \end{aligned} \tag{3.38}$$

where $p(X_1, X_2, ..., X_N)$ is equal, according to the conditional probability chain rule, to $p(X_1)$ $p(X_2/X_1)p(X_3/X_1, X_2)...p(X_N/X_1, X_2, X_3, ..., X_{N-1})$. If we assume the sequence as a i.i.d, then

$$p(X_1)p(X_2/X_1)...p(X_N/X_1, X_2, ..., X_{N-1}) = p(X_1)p(X_2)...p(X_N) \tag{3.39}$$

Then Eq. 3.38 becomes

$$\begin{aligned} H(X_1, X_2, ..., X_N) &= -\sum_{X_1} \sum_{X_2} ... \sum_{X_N} p(X_1, X_2, ..., X_N) \log_2 p(X_1, X_2, ..., X_N) \\ &= -\sum_{X_1} \sum_{X_2} ... \sum_{X_N} p(X_1)p(X_2)...p(X_N) \log_2 p(X_1)p(X_2)...p(X_N) \\ &= -\sum_{X_1} p(X_1) \sum_{X_2} p(X_2) ... \sum_{X_N} p(X_N) \log_2 p(X_1)p(X_2)...p(X_N) \end{aligned} \tag{3.40}$$

According to the rule properties of joint entropy:

$$\begin{aligned} H(X_1, X_2, ..., X_N) &\leq H(X_1) + H(X_2) + ... + H(X_N) \\ &= -\sum_{X_1} p(X_1) \log_2 p(X_1) + \sum_{X_2} p(X_2) \log_2 p(X_2) ... + \sum_{X_N} p(X_N) \log_2 p(X_N) \end{aligned} \tag{3.41}$$

Thus, the sum of the entropy of a sequence element is the upper bound of the entropy of a sequence. Now, finding the upper bound of each element entropy allows us to know the potential information in a series of measures.

The concave function $f(x) = -x \log_2 x$ follows the Jensen's inequality. That is a concave function $f(x)$ in an interval $[a, b]$ and $(y_1, y_2, .., y_i)$ are points in $[a, b]$ then :

$$n.f(\frac{(y_1 + y_2 + ... + y_i)}{n}) \geq f(y_1) + f(y_2) + ... + f(y_i) \tag{3.42}$$

Which implies:

$$-n.(\frac{p(x_1)+p(x_2)+...+p(x_i)}{n})\log_2\frac{(p(x_1)+p(x_2)+...+p(x_i))}{n} \geq \sum_{i=1}^{n} p(y_i)\log_2 p(y_i) \tag{3.43}$$

where x_i are the possible values of a one dimensional random variable X. Hence, x_i is a discrete random variable, then the sum pf probabilities is equal to 1.

$$-\log_2\frac{1}{n} \geq \sum_{i=1}^{n} p(y_i)\log_2 p(y_i) \tag{3.44}$$

$\log_2 n$ is nothing but the entropy of a homogeneous distribution. Therefore, the maximum entropy for a discrete random variable is $\log_2 n$, where n is the possible number of values for each random variable x.

In case of a sequence vector A with N elements, the maximum entropy is

$$\begin{aligned} H(A) &\leq \sum_{i=1}^{N} H(X_i) \\ &\leq \sum_{i=1}^{N} log(n) \end{aligned} \tag{3.45}$$

If A is a binary vector, which is the case for the FP techniques and the decision classifiers in general (e.g., Decision trees, XGB), the maximum entropy of a vector is $N\log_2 2$, where N is the length of the vector. In the case of a single precision number (32bits), the entropy is equal to $N\log_2 2^{32}$ which is 32 times greater. Therefore, a regularization function with float numbers provides extremely more information and allows us to learn a better regularization function. As an intuitive example, given a regularization technique that treats binary sequence, we have two sequences of preictal probability, O $\{0.43, 0.48, 0.49, 0.45, 0.5\}$ and L $\{0.1, 0.3, 0.01, 0.03, 0.2, 0.8\}$. Given a threshold 0.5, both sequences will have the same binary vector and will be treated equally by the regularizing function. The method we proposed, uses the probability output of a discriminator as input for another Machine Learning function that learns the pattern to optimize generation of alarms.

3.6 Model optimization and regularization

Deep Learning algorithms are formulated to solve an optimization problem, fundamentally, by applying gradient descent methods. It belongs to the parametric Machine Learning family, where the training goal is to learn from the dataset the parameters that achieve the minimum loss. The optimization algorithm itself is subject for optimization by controlling its hyper-parameters such as a learning rate, learning rate update, number of units and layers of the architecture, training time, batch size, etc. Deep neural networks are well known to solve non-linear optimization problems. Nevertheless, it is not considered a convex optimization, hence, it is highly subject to local optima. In optimization, local optima is an optimal solution with respect to the neighborhood but not optimal with respect to the whole parametric space. In Machine Learning, local optima corresponds to overfitting, where the model learns the optimal solution with respect to the training samples without being able to generalize. It is described as memorizing the examples instead of understanding the general pattern. Mitigating the problem of overfitting has also been considered as a part of model optimization. Numerous regularization techniques are proposed, such as L2 and L1 norm regularization, Dropout, Early-stop, and data augmentation. In this section, we introduce the most prominent approach applied in our work for model optimization.

3.6.1 Learning rate selection and adaptive optimization

The Stochastic Gradient Descent represents a variation of algorithms which are the main core of most Machine Learning methods. It consists of selecting stochastically a subset of the training data set and optimizing the network parameters according to the gradient of the objective function with respect to the network parameters as shown in Algorithm 1.

Learning rates control the step size of the weights update. If the learning rate is too small, the model will converge slowly with a high chance of falling into local minima. Alternatively, if the update step is big, the model will converge fast, yet, unstable and with a high chance of bouncing around the optimal point. Thus, it is recommended to train the model with different learning rates to find the best value for the model. One can combine the advantages of high and low learning rates by applying learning rate scheduled decay or by applying adaptive learning rates algorithms. The first method is based on the premise that a high learning rate guarantees fast convergence and avoids narrow local minima. When the loss function starts to oscillate or reach a plateau, then a smaller step update is needed to obtain a smooth convergence. Adaptive learning rate is a family of algorithms that adjusts the learning rates with respect to the loss function and the convergence rate, generally. It addresses the problem

Algorithm 1: Stochastic Gradient Descent (SGD)

Input: Training data S, regularization parameters λ, learning rate η, initialization σ
Output: Model parameters $\Theta = (w_0, \mathbf{w}, \mathbf{V})$
$w_0 \leftarrow 0$; $\mathbf{w} \leftarrow (0, \ldots, 0)$; $\mathbf{V} \sim \mathcal{N}(0, \sigma)$;
repeat
  for $(x, y) \in S$ **do**
    $w_0 \leftarrow w_0 - \eta(\frac{\partial}{\partial w_0} l(y(\mathbf{x} \mid \Theta), y) + 2\lambda^0 w_0)$;
    for $i \in \{1, \ldots, p\} \wedge x_i \neq 0$ **do**
      $w_i \leftarrow w_i - \eta(\frac{\partial}{\partial w_i} l(y(x \mid \Theta), y) + 2\lambda^w_\pi w_i)$;
      for $f \in \{1, \ldots, k\}$ **do**
        $v_{i,f} \leftarrow v_{i,f} - \eta(\frac{\partial}{\partial v_{i,f}} l(y(x \mid \Theta), y) + 2\lambda^v_{f,\pi(i)} v_{i,f})$;
      end
    end
  end
until *stopping criterion is not met*;

of the high sensitivity of a learning rate performance with respect to other hyperparameters (number of layers, number of units, activation function, etc.), and hindering the impact of the initial choice of learning rate. Considering the sparsity of the gradient matrix and its high variance, it is convenient to scale the learning rate according to the gradient of each parameter. In this study we considered two of the most prominent algorithms: ADAM [86] and RMSprop [87] optimization.

3.6.1.1 ADAM optimization

Adam optimization bases its approach on the adaptation of the learning rate for each parameter in the network. It models the gradient flow as a random variable where each new minibatch is used to update the mean and the uncentered variance following this equation:

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1 - \beta_1) g_t \\ v_t &= \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \end{aligned} \tag{3.46}$$

where m and v are the mean and uncentered variance, respectively. β_1 and β_2 are two introduced hyperparameters for adaptive rate control. The estimated mean and variance dimensions are equal to the number of parameters in the network. They scale the initial

learning rate by the simple equation

$$w_t = w_{t-1} - \eta \frac{m_t}{\sqrt{v_t} + \epsilon} \tag{3.47}$$

3.6.1.2 RMSprop

RMSprop is an improved version of the Rprop algorithm proposed by [88] which tackles the unsuitability of the Rprop with minibatch training. It has shown to be very efficient in recurrent neural networks. The RMSprop consists of computing a moving root mean square of the gradient for each minibatch and injecting it to scale the learning rate in the weight update equation.

$$\begin{aligned} E[g^2]_t &= \beta E[g^2]_{t-1} + (1-\beta) g_t^2 \\ w_t &= w_{t-1} - \eta \frac{g_t}{\sqrt{E[g^2]_t}} \end{aligned} \tag{3.48}$$

3.7 Regularization Methods

As highlighted above, Deep Learning models are prone to overfitting, especially in case of a small or invariant dataset. Formally, the gradient descent algorithm controls the value of the parameter according to the differentiation of the cost function with respect to itself. If the weight of a neural input is high, then its combined input becomes decisive. In the case of overfitting, the training process assigns high weights for some features that are specific to the training example and doesn't reflect the general distribution of an objective function. Therefore, it is highly recommended to regularize the weight's amplitude to avoid assigning to a single feature a decisive credit. Instead, with minimal sparsity in the weights matrix, it is guaranteed that the inference output is a collective decision that accredits different features as much as possible. In this section, we briefly introduce the methods we have adopted and those suitable for the nature of our datasets.

3.7.0.1 Weight Decay

The weight decay approach directly tackles the problem of overfitting by putting constraints on the weight matrix's magnitude. It assumes a prior distribution of the weights and optimizes the likelihood function accordingly. As a result, a gaussian and laplacian priori for the weight matrix is equivalent to minimizing the Euclidean and Taxicab norm, respectively. In

this study, we applied a gaussian weight regularization in the following loss function

$$L(y,p) = -[ylog(p) + (1-y)log(1-p)] + \lambda \|W\|_2^2 \tag{3.49}$$

3.7.0.2 Dropout

The Dropout techniques [89] applied a random binary mask on each layer of the network. The inputs of a layer L_i with N units is multiplied with a binomial multivariate random variable $\boldsymbol{x}$ that has the same dimension as L_i. As a result, each unit is deactivated with a probability p during the training phase (figure 3.6. The concept behind Dropout can be interpreted as a generalization of image random cropping in data augmentation. Instead of deleting random pixels from the image to avoid overfitting to a single feature, one can apply the same concept on the neural network features which are considered as deep features of the input. In our study, we selected $p = 0.5$ according to the recommendation of the author [89].

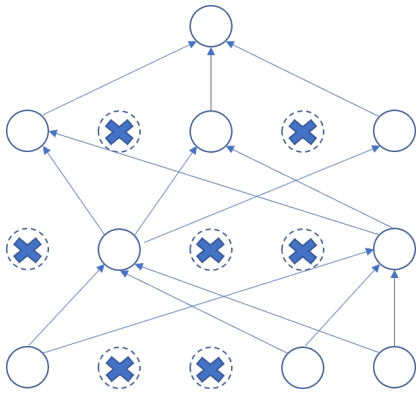

Figure 3.6 Illustration of dropout technique. During the training phase, in each layer, we deactivate each neuron with a binary probability P which prevent the model from depending on a limited subset of neurons

3.7.0.3 Early-stop

Another way of model regularization is to stop the training phase before the overfitting starts as illustrated in figure 3.7. One can randomly sample a small portion of the training samples and use them for validation during the training. Between every epoch, the model is tested on the validation portion . During the training, one should stop the training upon the early sign of overfitting which is reflected by the divergence between the performance on validation samples and the training ones.

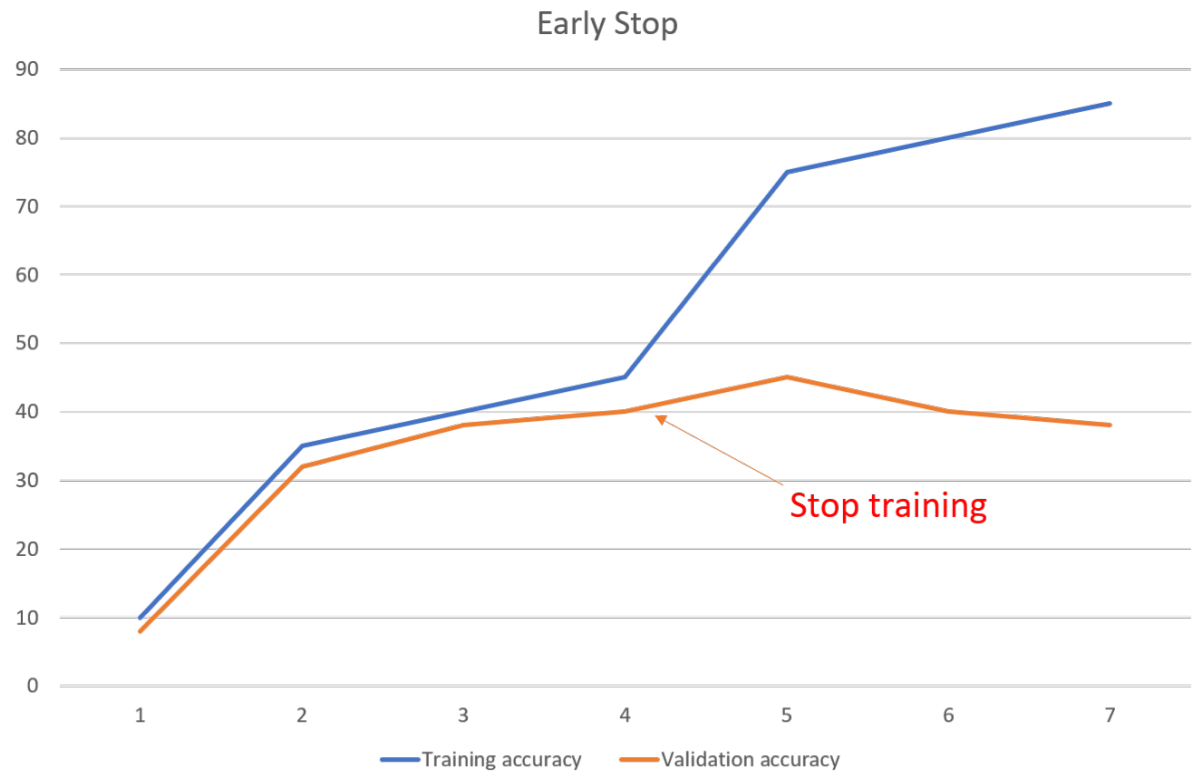

Figure 3.7 Illustration of early stop technique. The model iterates over the training dataset multiple time which can lead to overfitting to the training dataset. Overfitting can be early detected once the validation accuracy starts to diverge from the training accuracy, hence, early stop should be considered.

3.7.0.4 Hyperparameters optimization

Deep learning architectures are controlled through several types of hyperparameters such as learning rate, dropout rate, weight decay rate, number of layers, number of units, nad learning rate optimizer. The impact of one hyperparameter is highly dynamic and sensitive to the other ones. One can apply a grid search for all reasonable combinations of hyperparameters. Another approach is by doing a stochastic sampling from the hyperparameters space. In this study, we adopt a Markov Chain Monte Carlo MCMC approach which bases the current sample on the prior one.

3.8 Performance metrics and evaluation approach

In this study, the model, which is described as binary classification, is trained to discriminate between two classes. Additionally, the dataset is highly imbalanced and is dedicated

for medical diagnosis where the specificity and sensitivity of the descriminator are equally important. We introduce the performance metrics we adopted with a brief description.

3.8.1 Sensitivity

The sensitivity is a measure of the ratio of the actual true positive cases among the ones predicted as positive. It can also be represented as false negative rate where we measure the ratio of cases that have been falsely predicted as negative over the all negative prediction. The sum of the sensitivity and the false negative rate is always equal to 1. Formally, sensitivity is computed as follows:

$$Sensitivity = \frac{true\ positive}{true\ positive + false\ negative} \tag{3.50}$$

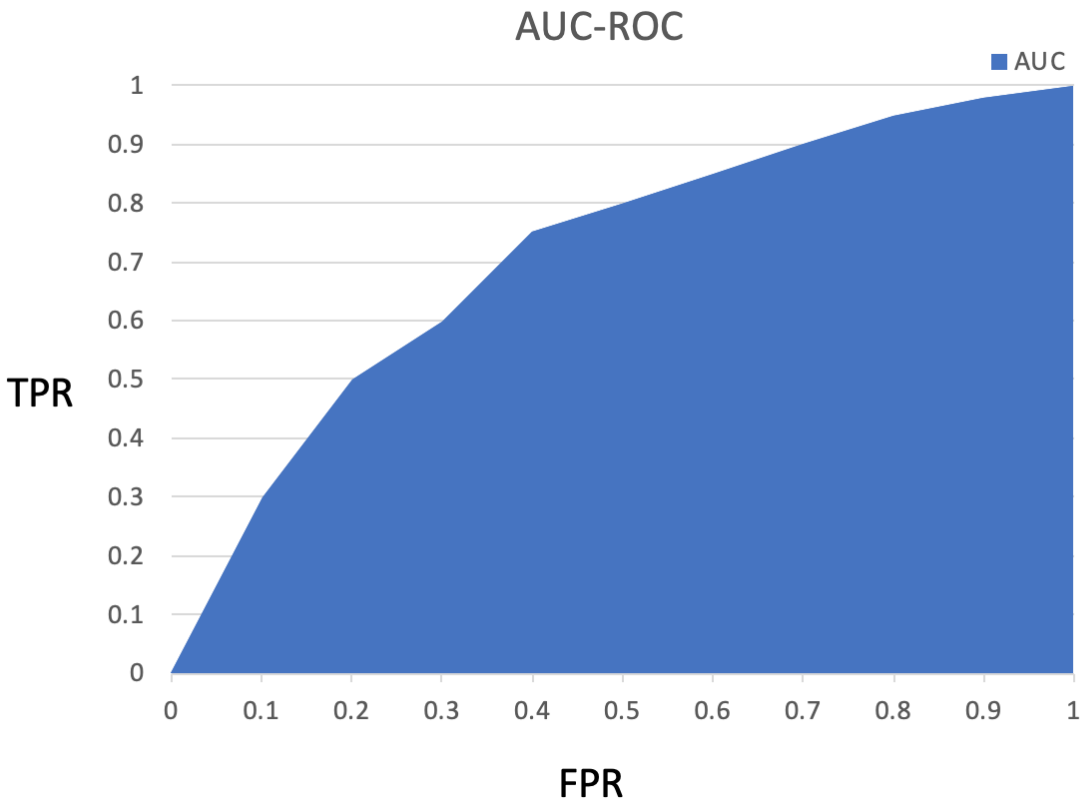

Figure 3.8 AUC-ROC curve. FPR: False Positive Rate; TPR: True Positive Rate. Each point in the graph represent the TPR and FPR of the model on a given threshold. By trying all possible thresholds, we obtain a curve where the blue color represent the area under curve. The area under curve reflects the level of discrimination between two distribution

3.8.2 False Positive Rate

The false positive rate (FPR), which can also be presented by specificity, reflects the accuracy in positive prediction. If the discriminator predicts a sample as positive, the false positive rate indicates the risk of obtaining an incorrect positive prediction among all positively predicted ones. Mathematically, it can be formulated as follows:

$$FPR = \frac{false\ negative}{false\ negative + true\ positive} \tag{3.51}$$

3.8.3 Area Under Curve

The performance of a discriminator is directly related to the decision threshold. The Area Under Curve of a Receiver Operating Characteristic curve (AUC-ROC) is a generic method that evaluates the potential of a discriminator independently of the decision threshold. First, we construct the ROC curve by computing the FPR and the True Positive Rate (FPR) of all possible threshold values as shown in figure 3.8. The interpolation of all points presents the ROC curve. The area under the ROC curve, practically, varies from 0.5 to 1 where 0.5 reflects a total random discrimination and 1 a zero overlap between two distribution.

3.9 Experiment design

The seizure prediction pipeline is primarily composed of signal preprocessing, features extraction and selection, training the classifier and regularization. As mentioned in Chapter 1, we aim to learn a regularization function from the dataset as an independent block, using Deep Learning architectures. Thus, our performance evaluation should focus on the improvement obtained by our regularization regardless of the classifier accuracy. For example, one could get high accuracy in some datasets irrespective of the regularization methods because of the compatibility between the dataset and the features/classifiers choice. Similarly, it is possible to have a low general performance that hides a proper regularization function. Therefore, we based our comparison between regularization methods on the amount of improvement over the classifier.

Our model follows the common pipeline described in figure 3.9. In the previous sections, we theoratically introduced each component of the algorithm, each apart. In the following, we present our detailed method for each phase in the algorithm and show our detailed experiment design:

3.9.1 Data acquisition

We collected the EEG recording from Epilepsy Ecosystem Melbourne Seizure Prediction dataset. Epilepsy Ecosystem is a crowd-sourcing ecosystem created to improve the state of the art of seizure prediction algorithms to make seizure prediction a practical treatment option for epilepsy patients [1]. The dataset is based on the three patients with worst seizure prediction performance in the Cook et al. study [36]. It aims to encourage the competitors to raise the lower bound of seizure prediction performance. Patient 1 is a 22-year-old female diagnosed with parietotemporat focal epilepsy. Her epilepsy was diagnosed at age 16. Patient 2 is a 51-year-old female. At age 10, she was diagnosed with occipitoparietal focal epilepsy. Patient 3 was a 50-year-old female, diagnosed with seizures of frontotemporal origin at age 15 [1].

Using the NeuroVista ambulatory monitoring device, iEEG data were recorded from humans with refractory focal epilepsy. Sixteen subdural electrodes were placed in each patient, targeting the seizure focus, and data were sampled at 400 Hz. To avoid any bias in the study, all seizures that occurred with a preceding seizure for less than four h were excluded, leaving just leading seizures in the dataset. Table 3.2 shows data characteristics for each patient.

Table 3.2 Data characteristics for the Epilepsy Ecosystem Melbourne Seizure Prediction Dataset

Patient	Recording duration (days)	Seizures	Lead seizures	Data clips (% interictal)
1	559	390	231	797 (69.0)
2	393	204	186	2027 (89.2)
3	374	545	216	2158 (88.3)

In our experiment, we adopted a segmentation window of 5 sec with 0% overlap. The total preictal segment is 1 hr equivalent to 1440000 samples ($400Hz * 3600sec$).

3.9.2 Data preprocessing and features extraction

In our study, the main concern is not related to the speed of prediction. Therefore, we adopted FIR filters for the preprocessing step. There are several types of FIR filters and to choose, one should consider the specifications that depend on many variables. Each filter method has its characterizations in pass-band, transition width, and stop-band. Table 3.1 displays the characteristics of the most prominent FIR filters. We applied a Chebyshev passband filter because of its narrow transition and monotonic stopband.

In most cases, electrodes recording the EEG signals can be different in terms of sensitivity and amplitude scale. Thus, exposing the classifier to an unbalanced voltage amplitude can impose a significant bias. To solve this, we normalized the amplitude of each channel signal by dividing each value by the maximum value. Next, we normalized it as described in Eq. 3.52 to obtain a zero centered distribution.

$$X_i = \frac{x_i - x_{mean}}{\sigma} \tag{3.52}$$

Our study investigates the regularization functions. Therefore, we select the most adopted features in studies with the best performance. We extracted the spectral power of the 5 EEG bands: Delta, Theta, Alpha, Beta, and Gamma. Additionally, we extracted the mean, standard deviation, kurtosis, and skewness. The aforementioned features have different scale; thus, we apply zero-mean normalization to avoid bias in the classification.

3.9.3 Classification

The dataset is split 80% for training and validation and 20% for testing. Each 5 sec EEG segment is represented by vector dimension equal 144, which equal to 16 (the number of leads) multiplied to 9 (number of features). We applied different classification method to elect the one with the highest AUC-ROC performance for each patient. The output of the trained classifier is a float number from 0 to 1 that reflects the probability of a preictal phase. The next Chapter present thorough details about the classifier training optimization.

3.9.4 Regularization

The dataset is composed of 10 min EEG record segments and every 6 consecutive preictal segments represent a complete preictal phase of one seizure episode. After training the classifiers, we align all segments in the temporal order and we map each 5 sec EEG window into a probability rate. As a result, we obtain a sequence of probabilities corresponding to each seizure episode. In other words, we take the interictal segments and preictal segments of each seizure episode and map it to a likelihood sequential vector. As explained in the previous section, the classifier outputs are prone to noisy alarms; thus, we apply different regularization method to investigate the best approach. First, we test the classical methods such as Firing Power technique and Kalman filter on each patient with different hyperparameters. For FP we tried threshold values that varies from 0.1 to 0,9 and window size that ranges from 5 time steps to 100. The KF was optimized with respect to T_p and K. Next, we train Deep Learning models such as one dimensional CNN, LSTM, BLSTM and MLP on learning to regularize

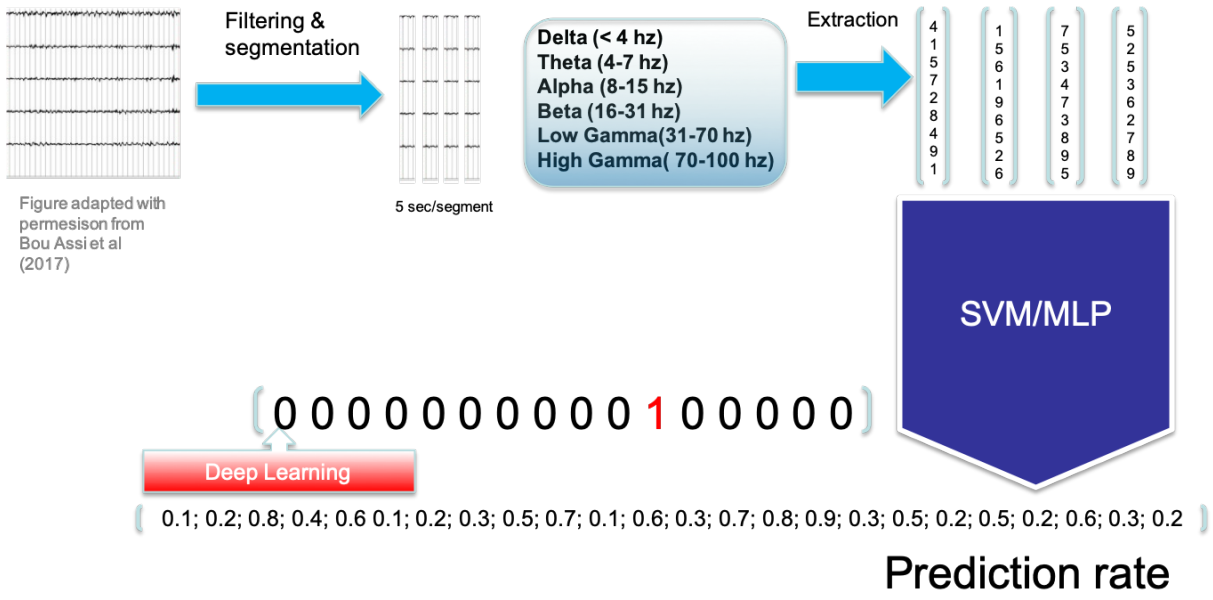

Figure 3.9 A schematic graph showing the general pipeline of our seizure prediction experiment. It starts with data filtering and segmentation. Next, we extract band power features and train our classifier which can be either SVM or MLP. Finally, we train different deep learning model to learn the regularizatoin function

the classifier output. Our study hypothesis assumes the possibility of learning an efficient and patient-specific regularization function from the EEG signal. To test our hypothesis, we compare the performance of the best Deep Learning model and other methods in term of specificity and sensitivity. To isolate our hypothesis from other factors, we should equally apply the same classifier on all compared regularization experiments. We could not compare our results to other authors work for the following reasons:

- The seizure prediction problem is not benchmarked, yet. Chapter 2 shows how state of the art is different from dataset to another. Our dataset has been, so far, used for crowdsourcing and competition where the classification task is applied for each EEG segment without a regularization step.
- Our experiments are highly dependent on classifier performance. The comparison with other regularization results is unreliable if the classifier is not shared and the dataset is not the same.
- We have found no published work applying machine learning to learn a post-processing regularization function. Therefore, our model can be perceived as a baseline model to be compared with in the future work in the field of learning regularization function.

I have read the draft and had a discussion with Shawn about the utility of pinning the PVCs in the center of the ECG frame. I am still not confident about some of the arguments concerning this issue. I am not sure what would be the best alternative. But I am curious to know what would be the result if the embedding was done on a one beat scale. Would the PVC samples be represented in a separate cluster? Regardless of the questions raised above, the current embedding holds a lot of potentials (since it is trained on a vast dataset). Hence, we can test it on supervised learning tasks on other datasets.

CHAPTER 4 RESULTS

4.1 Classifier results

The classifier is considered the core of the general seizure prediction pipeline. If the classifier's performance is weak, post processing regularization will not provide any advantage. Thus, we emphasize on the importance of exploiting all possible architectures to guarantee best discrimination results. Considering our proposed method for regularization, we opt for probabilistic classifiers which are able to generate a float number indicating the probability of a given class which has enormously more entropy than a binary output (subsection 3.5.1). We trained two different machine learning probabilistic classifiers: MLP and SVM. We ignored simple linear probabilistic classifiers based on our prior knowledge of the model complexity and non-linearity. All models are trained on each patient dataset where each dataset was split into 70% for training, 15% for validation and 15% for testing.

4.1.1 MLP

With the advancement of Deep Learning, fully connected networks have been controlled by a significant type of hyperparameters (number of layers, number of units, initial learning rate, L2 regularization rate, dropout rate, skip connection, activation function, etc.). This makes it very time expensive to explore all the combinations for all possible hyper parameters. We adopted ADAM algorithm for learning rate optimization which is less sensitive to the initial learning rate. Our learning rate initial value is 0.001. The number of units per layer and the number of layers, together, have a very high number of combination. Hence, we chose a fixed number of layers and experimented different number of units for each layers. It is important to note that our empirical findings revealed a higher impact for the number of units compared to the number of layers. The dropout rate is a very sensitive parameter and there is no method to define its best value; thus, we included it in our grid search.

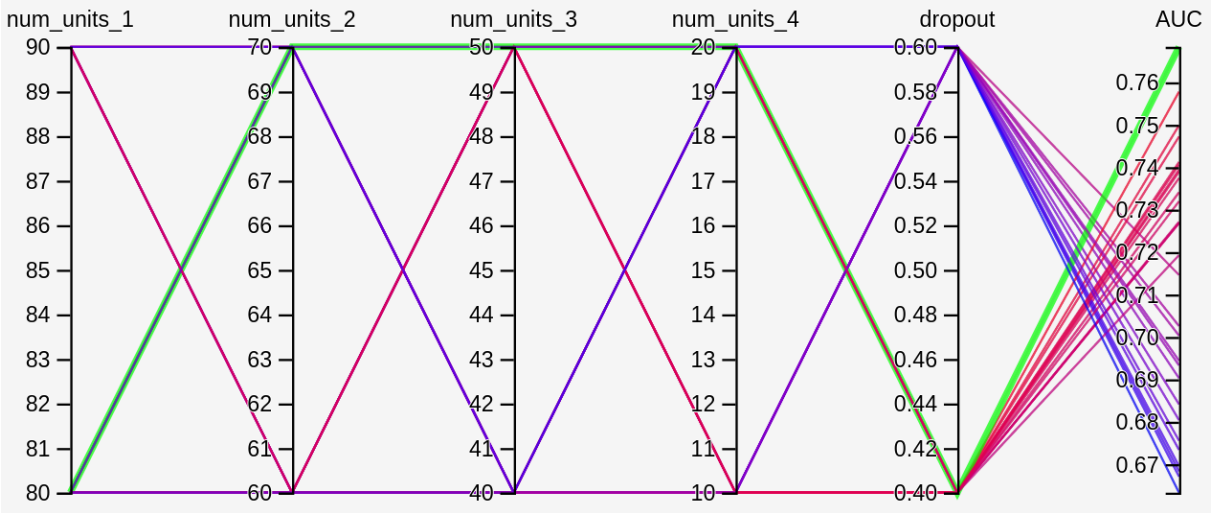

Figure 4.1 Parallel coordinates of the hyperparameters research for MLP in Patient1. Each hyperparameter combination is represented with one colored line that maps the parameters values to their correspondent performance (AUC-ROC). The green line shows the best hyperparameters in our grid-search.

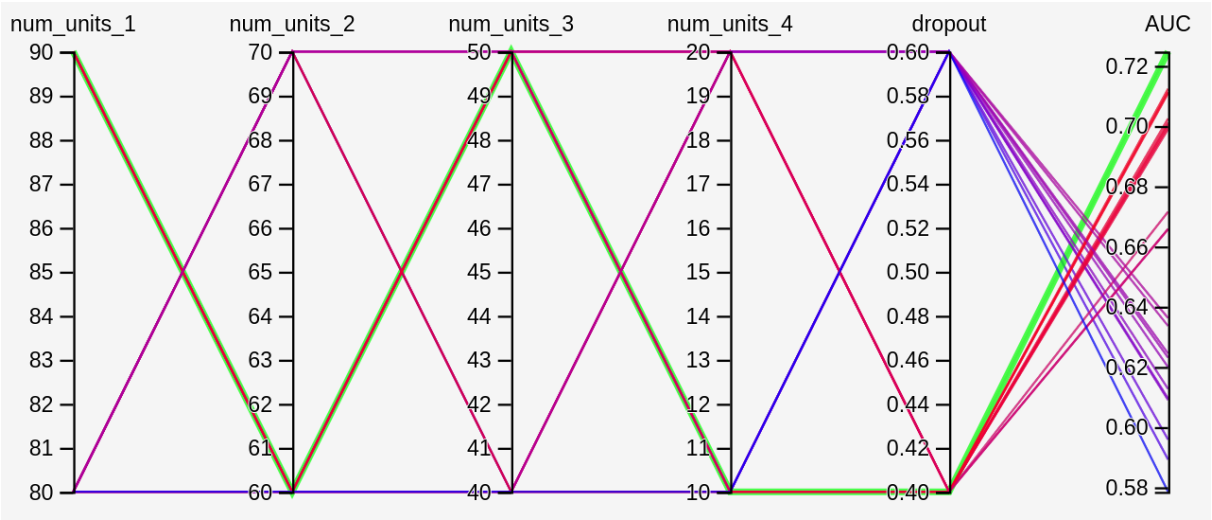

Figure 4.2 Parallel coordinates of the hyperparameters research for MLP in Patient2. Each hyperparameter combination is represented with one colored line that maps the parameters values to their correspondent performance (AUC-ROC). The green line shows the best hyperparameters in our grid-search.

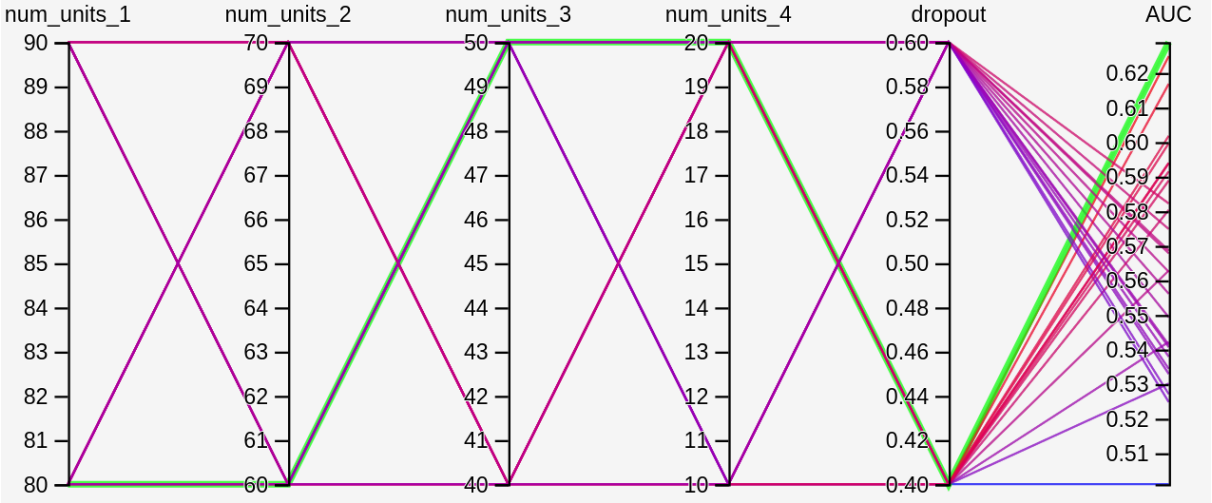

Figure 4.3 Parallel coordinates of the hyperparameters research for MLP in Patient3. Each hyperparameter combination is represented with one colored line that maps the parameters values to their correspondent performance (AUC-ROC). The green line shows the best hyperparameters in our grid-search.

Figures 4.1, 4.2 and 4.3 show our grid search results for each individual. We chose hyperparamter values that correspond to the highest AUC score. If the best hyperparameter combinations are outliers corresponding to their neighbours, we ignore them and adopt the combination with the consistent performance. In all patients, the optimum performance was obtained with Adam optimizer, dropout rate equal to 0.4. The optimum number of units for each layer varied from patient to another; however our findings show that the contractive architecture (smaller number of units) has a better performance.

4.1.2 SVM

SVM is, basically, a decision classifier and its architecture does not provide output probability. Considering the efficiency of SVM in preictal discrimination, we applied a sigmoid function to the distance between the sample and the hyper-plane boundary to obtain a probabilistic prediction. Compared to MLP, SVM has less hyperparamters to control. There are several types of kernel functions in SVM, however, we only applied the Gaussian kernels because of its high reputation in seizure prediction. The SVM models were tuned by gamma and C. Gamma represents the kernel scaling parameters and C represents the penalty parameter of the error term. Figures 4.4, 4.5 and 4.6 show our grid search for the best hyperparameters. Similar to MLP we dropped the outlier combinations. Table 4.1 presents each classifier type and the best AUC-ROC obtained after parameters optimization.

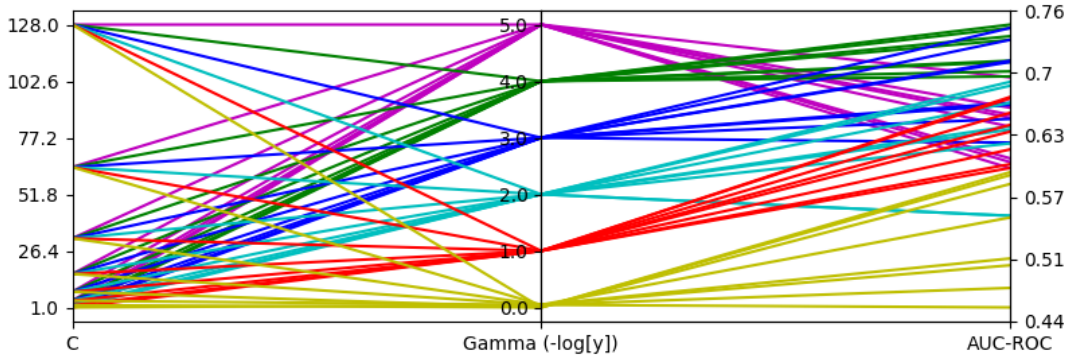

Figure 4.4 Parallel coordinates of the hyperparameter research for SVM in Patient1. Gamma represents the kernel scaling parameters and C represents the penalty parameter of the error term. Each Gamma value is presented with different color to highlight the impact of gamma on the final performance

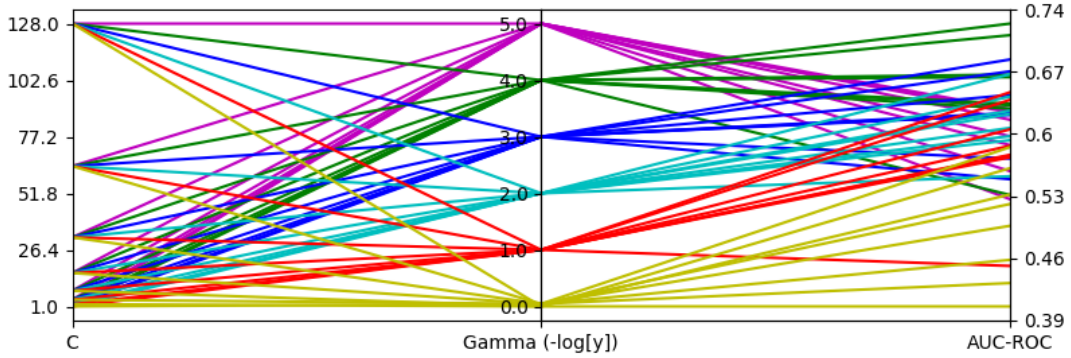

Figure 4.5 Parallel coordinates of the hyperparameter research for SVM in Patient2

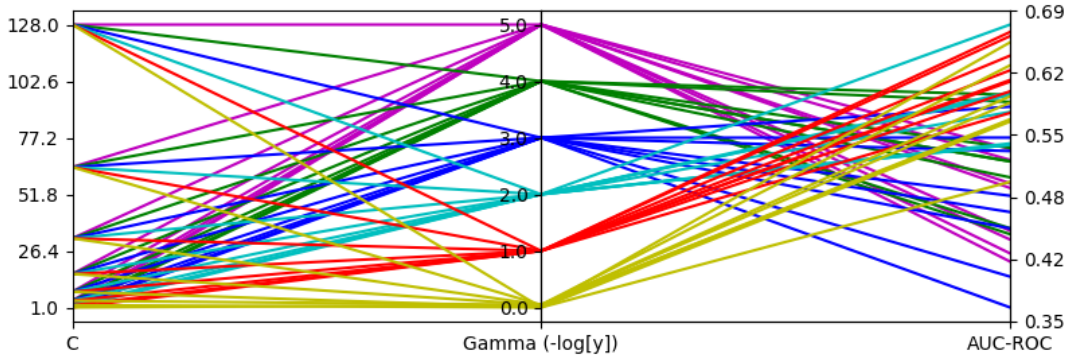

Figure 4.6 Parallel coordinates of the hyperparameter research for SVM in Patient3

Table 4.1 AUC-ROC for MLP and SVM in each Patient

Patient	MLP	SVM
Patient 1	**0.7701**	0.7468
Patient 2	0.7133	**0.7432**
Patient 3	0.5942	**0.6928**

4.2 Regularization results

The probabilistic classifier output is a one dimension float number that ranges from 0 to 1 and represents the probability of entering a preictal phase. To explore the capacity of Deep Learning to learn a regularization function, we experimented four different neural network architectures: one dimensional CNN, LSTM, BLSTM and MLP. Each model was trained independently with respect to the main hyperparameters (Table 4.2). Except for the LSTM, where the number of hidden units are chosen independently of the input size, the number of units have been chosen with respect to each window size. The window size is the number of prediction outputs the regularizer takes as input.

Table 4.2 Optimized model hyperparameters for different Deep Learning architectures

Architecture	Layers	Units	Feature maps	LR optimizers	Dropout
MLP	4	—	—	ADAM	0.6
CNN	3	—	62-32	ADAM	0.5
LSTM	1	24	—	RMS	0.5
BLSTM	1	24	—	RMS	0.5

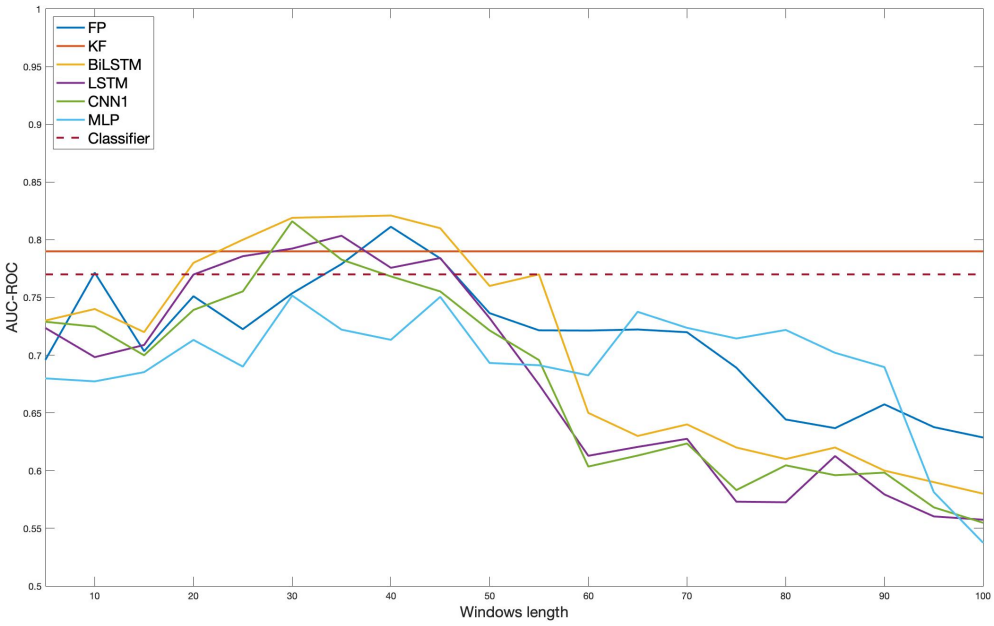

Figure 4.7 Patient 1 results for different models as a function of window size. This graph aims to emphasize the role of the regularization window size in the overall performance

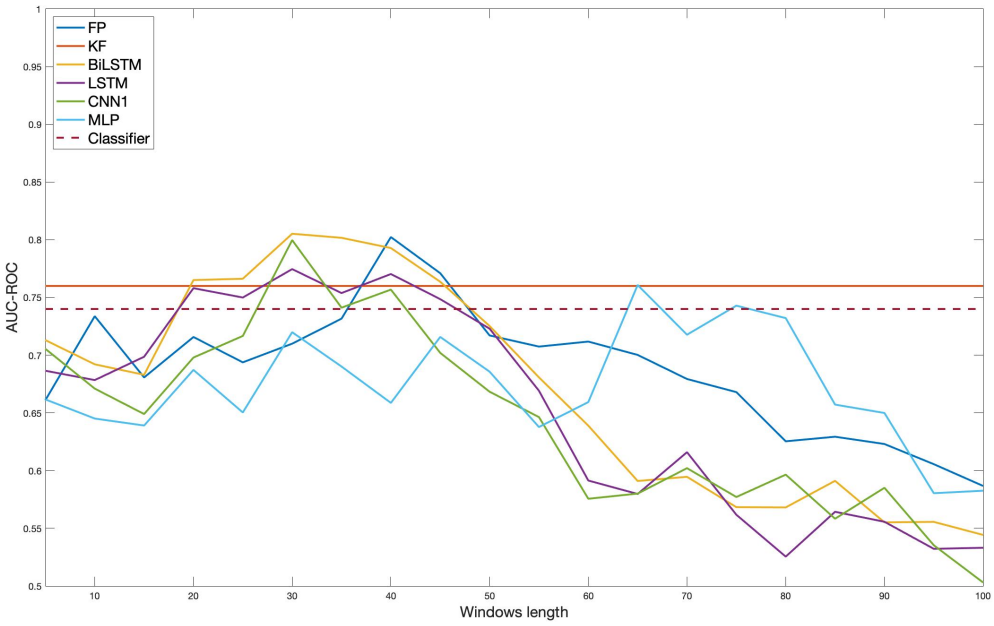

Figure 4.8 Patient 2 results for different models as a function of window size

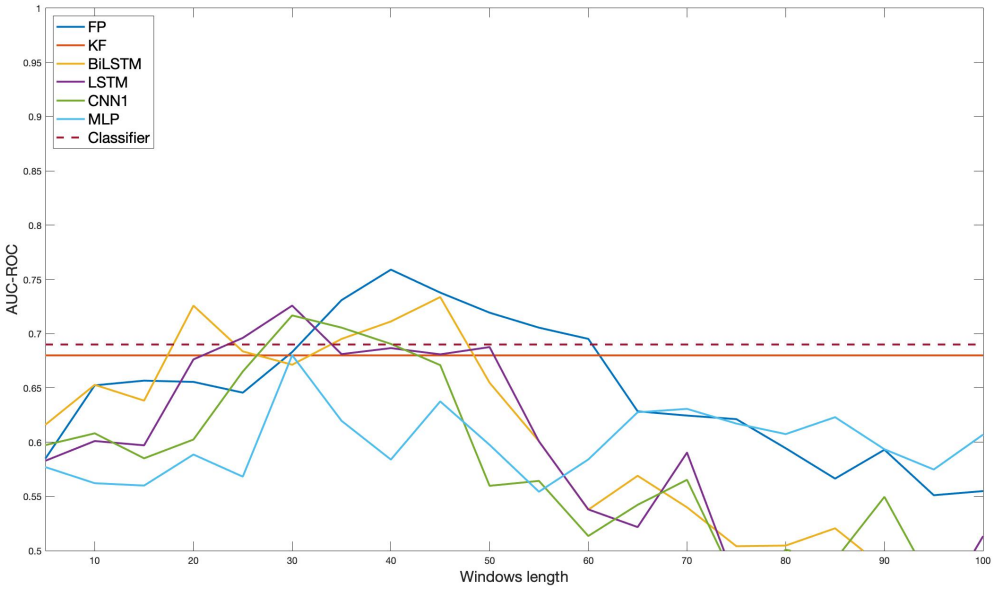

Figure 4.9 Patient 3 results for different models as a function of window size

The FP method depends on the decision threshold and the window size. At the first stage we adopted the AUC-ROC metric to obtain a generalized comparison independent of the threshold value. We compared the AUC-ROC to each window considering the key impact of the window size on the model itself and on the size of the training dataset. The KF depends on the prediction function and the Kalman gain. Figures 4.7, 4.8 and 4.9 show the performance of different types of regularizers in different window lengths. The impact of the window size appears to be critical on the regularizer performance. Narrow windows are more sensitive to false alarms which implies less information about the positive prediction patterns. On the other hand, wide windows ($>$ than 70) have also a negative impact on performance. Although a wide window provides more information, it dissolves the weight of a burst of alarms and decreases the sensitivity of a classifier. Substantially, Deep Learning models suffer from the shortage of training samples in case of large windows. Approximately, the best window size is considered between 20 and 60 time steps. Considering Deep learning models, BLSTM predominates all trained models in all patients. It recursively processes the input with a long term memory in two directions which align with the sequential aspect of the EEG signal. The KF is independent of the window size and it only considers the previous sample and its derivative. It has a poor performance, in general, and in patient 3 the classifier alone is more accurate without a KF regularization. The main interpretation is that KFs only consider one sample before processing and the prediction function for seizure

prediction is not based on any study of the dynamical progress of the preictal phase. We simply approached the prediction function by assuming a monotonic and linear increase of the preictal phase with each time step.

Table 4.3 Best performance for all models (AUC-ROC/window size)

Patient	Classifier	MLP	CNN	LSTM	BLSTM	FP	KF
Patient 1	0.770/—	0.751/30	0.816/30	0.803/35	**0.821**/40	0.817/40	0.791/—
Patient 2	0.743/—	0.762/80	0.779/30	0.774/45	**0.805**/30	0.802/40	0.762/—
Patient 3	0.692/—	0.62/85	0.716/30	0.725/30	0.733/45	**0.759**/40	0.681/—

Table 4.3 highlights the best results and window size for each model. The BLSTM architecture is consistently performing superior to other Deep Learning models which is probably due to the reasons mentioned above. FP has similarly robust results despite the fact that it is a non learned linear function. For a further comparison of the performance between FP, KF and Learning models, we measured the sensitivity and the specificity score for the best deep learning model (BLSTM), FP and KF. Table 4.4 shows the sensitivity and the specificity score where the decision threshold has the highest F1 score. As the table highlights, our deep learning model shows no less specificity compared to FP methods, except in Patient 3 where the discriminative classifier has already a significant underperformance. Deep learning models are very strong in capturing an underlying pattern for a sequence of inputs. In Patient 3, we assume that the preictal phase progression is more stochastic and harder to learn. However, our findings confirm that a learned regularization function can learn more than a simple filtering task and can maintain a high sensitivity rate compared to classical methods. In Patient 1, our findings show that BLSTM did not only maintain the sensitivity when increasing the specificity but also improved the sensitivity of the model which proves that the model is capable of learning the dynamics and the pattern of preictal seizure probability. The performance of BLSTM shown in Figure 4.7, 4.8 and 4.9 along with the Table 4.4 demonstrates a robustness and consistency which, combined with the high accuracy, make it an improved replacement of FP and KF.

Table 4.4 Comparison of the sensitivity and the specificity between KF, FP and BLSTM based on the best threshold value for each method. SS: sensitivity; SP: specificity; AUC: AUC-ROC

Model	FP		BLSTM		KF		Classifier	
Metrics	SS	SP	SS	SP	SS	SP	SS	SP
Patient 1	0.61	0.93	**0.69**	**0.94**	0.64	0.82	0.67	0.79
Patient 2	0.59	**0.92**	**0.62**	**0.92**	0.58	0.85	0.64	0.87
Patient 3	**0.52**	**0.87**	0.49	0.78	0.46	0.72	0.61	0.76

Our results suffer from some limitations. The dataset has been mainly used for competition, and there are no published studies with a detailed methodology to compare our results. The leaderboard score is standardized with respect to the EEG segment length, which is determined by the competition organizers. Additionally, only a small portion of the test dataset is public and, therefore, there is no access to the private testing samples from outside the competition. Furthermore, the research question we proposed is restricted to the regularization techniques which, as a single stage in the prediction pipeline, can only be compared to other methods if we are using the exact same features and classifier.

KFs, as our empirical results show, are not able to regularize an extremely non-linear signal like EEG signal. Nevertheless, our findings cannot be generalized over KFs in general, since there are many variations that are suited for non-linear dynamics and with a second order estimation. Extended KFs can approximate non-linear functions to some extent. Nevertheless, the core limitation in KFs is in the ignorance of the state prediction equation. There are no evidence, so far, that assumes that the probability of a preictal phase increases monotonically and linearly. Not to mention, the dynamical progress is highly dependent on each patient which endorses the advantage of learning the dynamics from each patient data.

CHAPTER 5 CONCLUSION

The general aim of regularization has been always to reduce the probability of false alarms by simply ruling out positive predictions that are out of the context. Classical methods had represented the context in various ways. The FP technique represents the context by the average sum value of a binary prediction within a time window. On the other hand, The KF considers one or two previous values with their derivatives in addition to the internal prediction equation that estimates the dynamics of a system. It represents the context as a one step quadratic estimation. The FP technique is robust and less sensitive to noises in general, especially in the case of large windows. Additionally it can be easily implemented on the top of the decision classifier family which is a wide group of algorithms (e.g., Decision Tree, Gradient Boost, K Nearest Neighbors). Yet, It is completely blind toward the pattern and the progression of the preictal state. KFs are better than FP in capturing the pattern; nevertheless, their capacity to capture a nonlinear pattern is very limited and they suffer from a short memory. Furthermore, KFs rigorously depend on the prediction equation that estimates the next state of a model which is not mathematically available, yet.

5.1 Summary of Works

In this study, we propose an alternative way of approaching the regularization step. We showed that with enough data samples, one can teach an artificially intelligent model to filter and regularize in an optimized and customized way. We also showed that the choice of the model is critical and the architecture should be structurally aligned with the input characteristics (in our case sequential with a long term dependency). We optimally trained different neural network architectures from different families and found the recurrent neural networks family to be the most suitable for the regularization task. We ended up by comparing the performance between classical methods and trained models and we showed the importance of the window size during regularization. Additionally, we elaborated the contribution of trained models by highlighting their capacity to maintain a relatively high sensitivity while remarkably increasing the specificity.

5.2 Limitations

Despite the outperformance of the Deep Learning model, we conclude that learned models are sensitive to the size of the dataset and the underlying performance of the discriminator.

Our study lacks detailed investigation for the reason behind the bad performance of BLSTM in Patient 3. Further investigations would highlight the chaosity in Patient 3 and study the impact of input randomness on the performance of intelligent models. Our study investigated the simplest version of KFs which limits our generalizations and conclusions, while, extended KFs can still approximate non-linear functions to some extent.

5.3 Future Research

This study opened the door to learn the post-processing regularization using patient's EEG record. The implied premise is that the reliability of a present probabilistic predition of a classifier is highly conditioned on the previous predictions. Our proposed method suggests learning the aforementioned conditional function separately and in two stages. For the future, we aim to study the possibility of embedding the regularization step in the classification itself. One could train the classifier to predict based on the present signal features and the previous predictions. This can teach the model to predict considering the prior prediction patterns. This has an advantage over recurrent neural networks in being less hungry to data.

Seizure prediction is a very important application of Machine Learning, yet, is very dangerous with respect to the risks associated to any failure of the system. The regularization aim is to hinder the unpredictability of Machine Learning models, especially, those that are not interpretable like ANNs. FP, for example, can be considered a security tool against wrong prediction caused by random noises in the signal. If a Deep Learning model is replacing FP, then it needs to be noise proof which can be achieved by applying a robust training that trains the model to predict correctly independently of every possible noise.

BIBLIOGRAPHY

[1] L. Kuhlmann, P. Karoly, D. R. Freestone, B. H. Brinkmann, A. Temko, A. Barachant, F. Li, G. Titericz Jr, B. W. Lang, D. Lavery *et al.*, "Epilepsyecosystem. org: crowd-sourcing reproducible seizure prediction with long-term human intracranial eeg," *Brain*, vol. 141, no. 9, pp. 2619–2630, 2018.

[2] U. Herwig, P. Satrapi, and C. Schönfeldt-Lecuona, "Using the international 10-20 eeg system for positioning of transcranial magnetic stimulation," *Brain topography*, vol. 16, no. 2, pp. 95–99, 2003.

[3] Y. Nagahama, A. J. Schmitt, D. Nakagawa, A. S. Vesole, J. Kamm, C. K. Kovach, D. Hasan, M. Granner, B. J. Dlouhy, M. A. Howard *et al.*, "Intracranial eeg for seizure focus localization: evolving techniques, outcomes, complications, and utility of combining surface and depth electrodes," *Journal of Neurosurgery*, vol. 1, no. aop, pp. 1–13, 2018.

[4] L. Kuhlmann, K. Lehnertz, M. P. Richardson, B. Schelter, and H. P. Zaveri, "Seizure prediction—ready for a new era," *Nature Reviews Neurology*, p. 1, 2018.

[5] M. Kim and A. Anpalagan, "Tor traffic classification from raw packet header using convolutional neural network," in *2018 1st IEEE International Conference on Knowledge Innovation and Invention (ICKII)*. IEEE, 2018, pp. 187–190.

[6] P. N. Banerjee, D. Filippi, and W. A. Hauser, "The descriptive epidemiology of epilepsy—a review," *Epilepsy research*, vol. 85, no. 1, pp. 31–45, 2009.

[7] I. Megiddo, A. Colson, D. Chisholm, T. Dua, A. Nandi, and R. Laxminarayan, "Health and economic benefits of public financing of epilepsy treatment in india: An agent-based simulation model," *Epilepsia*, vol. 57, no. 3, pp. 464–474, 2016.

[8] W. A. Hauser and L. T. Kurland, "The epidemiology of epilepsy in rochester, minnesota, 1935 through 1967," *Epilepsia*, vol. 16, no. 1, pp. 1–66, 1975.

[9] J. W. Sander, "The epidemiology of epilepsy revisited," *Current opinion in neurology*, vol. 16, no. 2, pp. 165–170, 2003.

[10] R. S. Fisher, C. Acevedo, A. Arzimanoglou, A. Bogacz, J. H. Cross, C. E. Elger, J. Engel Jr, L. Forsgren, J. A. French, M. Glynn *et al.*, "Ilae official report: a practical clinical definition of epilepsy," *Epilepsia*, vol. 55, no. 4, pp. 475–482, 2014.

[11] H. Stefan and F. H. Lopes Da Silva, "Epileptic neuronal networks: methods of identification and clinical relevance." *Frontiers in neurology*, vol. 4, p. 8, 2013.

[12] C. P. Panayiotopoulos, "Typical absence seizures and related epileptic syndromes: assessment of current state and directions for future research," *Epilepsia*, vol. 49, no. 12, pp. 2131–2139, 2008.

[13] K. J. Werhahn, S. Noachtar, S. Arnold, M. Pfänder, A. Henkel, P. A. Winkler, and H. O. Lüders, "Tonic seizures: their significance for lateralization and frequency in different focal epileptic syndromes," *Epilepsia*, vol. 41, no. 9, pp. 1153–1161, 2000.

[14] P. L. Nunez, R. Srinivasan *et al.*, *Electric fields of the brain: the neurophysics of EEG*. Oxford University Press, USA, 2006.

[15] C. J. Chu, "High density eeg—what do we have to lose?" *Clinical neurophysiology: official journal of the International Federation of Clinical Neurophysiology*, vol. 126, no. 3, p. 433, 2015.

[16] V. Jurcak, D. Tsuzuki, and I. Dan, "10/20, 10/10, and 10/5 systems revisited: their validity as relative head-surface-based positioning systems," *Neuroimage*, vol. 34, no. 4, pp. 1600–1611, 2007.

[17] J. W. Britton, L. C. Frey, J. Hopp, P. Korb, M. Koubeissi, W. Lievens, E. Pestana-Knight, and E. L. St, *Electroencephalography (EEG): An introductory text and atlas of normal and abnormal findings in adults, children, and infants*. American Epilepsy Society, Chicago, 2016.

[18] P. Agante and J. M. De Sá, "Ecg noise filtering using wavelets with soft-thresholding methods," in *Computers in Cardiology 1999. Vol. 26 (Cat. No. 99CH37004)*. IEEE, 1999, pp. 535–538.

[19] E. B. Assi and S. Rihana, "Kmeans-ica based automatic method for eog denoising in multi-channel eeg recordings," in *Proceedings of 11th IASTED International Conference on Biomedical Engineering, Zurich*, 2014, pp. 23–25.

[20] W. A. Hauser and E. Beghi, "First seizure definitions and worldwide incidence and mortality," *Epilepsia*, vol. 49, pp. 8–12, 2008.

[21] D. J. Chong and A. M. Lerman, "Practice update: review of anticonvulsant therapy," *Current neurology and neuroscience reports*, vol. 16, no. 4, p. 39, 2016.

[22] P. Kwan and M. J. Brodie, "Early identification of refractory epilepsy," *New England Journal of Medicine*, vol. 342, no. 5, pp. 314–319, 2000.

[23] M. R. Sperling, M. J. O'connor, A. J. Saykin, and C. Plummer, "Temporal lobectomy for refractory epilepsy," *Jama*, vol. 276, no. 6, pp. 470–475, 1996.

[24] R. George, A. Sonnen, A. Upton, M. Salinsky, R. Ristanovic, D. Bergen, W. Mirza, W. Rosenfeld, D. Nari-Toku, R. Manon-Espaillat *et al.*, "A randomized controlled trial of chronic vagus nerve stimulation for treatment of medically intractable seizures," *Neurology*, vol. 45, no. 2, pp. 224–230, 1995.

[25] W. H. Theodore and R. S. Fisher, "Brain stimulation for epilepsy," *The Lancet Neurology*, vol. 3, no. 2, pp. 111–118, 2004.

[26] A. T. Tzallas, M. G. Tsipouras, D. G. Tsalikakis, E. C. Karvounis, L. Astrakas, S. Konitsiotis, and M. Tzaphlidou, "Automated epileptic seizure detection methods: a review study," in *Epilepsy-histological, electroencephalographic and psychological aspects*. IntechOpen, 2012.

[27] F. Mormann, T. Kreuz, C. Rieke, R. G. Andrzejak, A. Kraskov, P. David, C. E. Elger, and K. Lehnertz, "On the predictability of epileptic seizures," *Clinical neurophysiology*, vol. 116, no. 3, pp. 569–587, 2005.

[28] E. B. Assi, D. K. Nguyen, S. Rihana, and M. Sawan, "Towards accurate prediction of epileptic seizures: A review," *Biomedical Signal Processing and Control*, vol. 34, pp. 144–157, 2017.

[29] Z. Rogowski, I. Gath, and E. Bental, "On the prediction of epileptic seizures," *Biological cybernetics*, vol. 42, no. 1, pp. 9–15, 1981.

[30] F. Mormann, R. G. Andrzejak, C. E. Elger, and K. Lehnertz, "Seizure prediction: the long and winding road," *Brain*, vol. 130, no. 2, pp. 314–333, 2006.

[31] K. Gadhoumi, J.-M. Lina, F. Mormann, and J. Gotman, "Seizure prediction for therapeutic devices: A review," *Journal of neuroscience methods*, vol. 260, pp. 270–282, 2016.

[32] J. Rasekhi, M. R. K. Mollaei, M. Bandarabadi, C. A. Teixeira, and A. Dourado, "Preprocessing effects of 22 linear univariate features on the performance of seizure prediction methods," *Journal of neuroscience methods*, vol. 217, no. 1-2, pp. 9–16, 2013.

[33] C. A. Teixeira, B. Direito, M. Bandarabadi, M. Le Van Quyen, M. Valderrama, B. Schelter, A. Schulze-Bonhage, V. Navarro, F. Sales, and A. Dourado, "Epileptic seizure predictors based on computational intelligence techniques: A comparative study with 278 patients," *Computer methods and programs in biomedicine*, vol. 114, no. 3, pp. 324–336, 2014.

[34] M. Bandarabadi, C. A. Teixeira, J. Rasekhi, and A. Dourado, "Epileptic seizure prediction using relative spectral power features," *Clinical Neurophysiology*, vol. 126, no. 2, pp. 237–248, 2015.

[35] M. Ihle, H. Feldwisch-Drentrup, C. A. Teixeira, A. Witon, B. Schelter, J. Timmer, and A. Schulze-Bonhage, "Epilepsiae–a european epilepsy database," *Computer methods and programs in biomedicine*, vol. 106, no. 3, pp. 127–138, 2012.

[36] M. J. Cook, T. J. O'Brien, S. F. Berkovic, M. Murphy, A. Morokoff, G. Fabinyi, W. D'Souza, R. Yerra, J. Archer, L. Litewka *et al.*, "Prediction of seizure likelihood with a long-term, implanted seizure advisory system in patients with drug-resistant epilepsy: a first-in-man study," *The Lancet Neurology*, vol. 12, no. 6, pp. 563–571, 2013.

[37] J. J. Howbert, E. E. Patterson, S. M. Stead, B. Brinkmann, V. Vasoli, D. Crepeau, C. H. Vite, B. Sturges, V. Ruedebusch, J. Mavoori *et al.*, "Forecasting seizures in dogs with naturally occurring epilepsy," *PloS one*, vol. 9, no. 1, p. e81920, 2014.

[38] Y. Park, L. Luo, K. K. Parhi, and T. Netoff, "Seizure prediction with spectral power of eeg using cost-sensitive support vector machines," *Epilepsia*, vol. 52, no. 10, pp. 1761–1770, 2011.

[39] J. J. Niederhauser, R. Esteller, J. Echauz, G. Vachtsevanos, and B. Litt, "Detection of seizure precursors from depth-eeg using a sign periodogram transform," *IEEE Transactions on Biomedical Engineering*, vol. 50, no. 4, pp. 449–458, 2003.

[40] M. Le Van Quyen, J. Martinerie, M. Baulac, and F. Varela, "Anticipating epileptic seizures in real time by a non-linear analysis of similarity between eeg recordings," *Neuroreport*, vol. 10, no. 10, pp. 2149–2155, 1999.

[41] M. Bandarabadi, J. Rasekhi, C. A. Teixeira, M. R. Karami, and A. Dourado, "On the proper selection of preictal period for seizure prediction," *Epilepsy & Behavior*, vol. 46, pp. 158–166, 2015.

[42] M. Larmuseau, "Epileptic seizure prediction using deep learning," 2016.

[43] N. Moghim and D. W. Corne, "Predicting epileptic seizures in advance," *PloS one*, vol. 9, no. 6, p. e99334, 2014.

[44] K. Gadhoumi, J.-M. Lina, and J. Gotman, "Seizure prediction in patients with mesial temporal lobe epilepsy using eeg measures of state similarity," *Clinical Neurophysiology*, vol. 124, no. 9, pp. 1745–1754, 2013.

[45] A. Aarabi, R. Fazel-Rezai, and Y. Aghakhani, "Eeg seizure prediction: measures and challenges," in *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2009, pp. 1864–1867.

[46] T. Netoff, Y. Park, and K. Parhi, "Seizure prediction using cost-sensitive support vector machine," in *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2009, pp. 3322–3325.

[47] L. D. Iasemidis, J. C. Sackellares, H. P. Zaveri, and W. J. Williams, "Phase space topography and the lyapunov exponent of electrocorticograms in partial seizures," *Brain topography*, vol. 2, no. 3, pp. 187–201, 1990.

[48] M. Le Van Quyen, J. Soss, V. Navarro, R. Robertson, M. Chavez, M. Baulac, and J. Martinerie, "Preictal state identification by synchronization changes in long-term intracranial eeg recordings," *Clinical Neurophysiology*, vol. 116, no. 3, pp. 559–568, 2005.

[49] M. Winterhalder, B. Schelter, T. Maiwald, A. Brandt, A. Schad, A. Schulze-Bonhage, and J. Timmer, "Spatio-temporal patient–individual assessment of synchronization changes for epileptic seizure prediction," *Clinical neurophysiology*, vol. 117, no. 11, pp. 2399–2413, 2006.

[50] F. Mormann, R. G. Andrzejak, T. Kreuz, C. Rieke, P. David, C. E. Elger, and K. Lehnertz, "Automated detection of a preseizure state based on a decrease in synchronization in intracranial electroencephalogram recordings from epilepsy patients," *Physical Review E*, vol. 67, no. 2, p. 021912, 2003.

[51] H. C. Lee, M. H. Kohrman, K. E. Hecox, and W. van Drongelen, "Seizure prediction," in *Neural Engineering*. Springer, 2013, pp. 685–723.

[52] P. Mirowski, D. Madhavan, Y. LeCun, and R. Kuzniecky, "Classification of patterns of eeg synchronization for seizure prediction," *Clinical neurophysiology*, vol. 120, no. 11, pp. 1927–1940, 2009.

[53] M. Bandarabadi, C. A. Teixeira, B. Direito, and A. Dourado, "Epileptic seizure prediction based on a bivariate spectral power methodology," in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2012, pp. 5943–5946.

[54] L. D. Coles, E. E. Patterson, W. D. Sheffield, J. Mavoori, J. Higgins, B. Michael, K. Leyde, J. C. Cloyd, B. Litt, C. Vite *et al.*, "Feasibility study of a caregiver seizure alert system in canine epilepsy," *Epilepsy research*, vol. 106, no. 3, pp. 456–460, 2013.

[55] C. Ding and H. Peng, "Minimum redundancy feature selection from microarray gene expression data," *Journal of bioinformatics and computational biology*, vol. 3, no. 02, pp. 185–205, 2005.

[56] L. Davis, "Handbook of genetic algorithms," 1991.

[57] B. Direito, F. Ventura, C. Teixeira, and A. Dourado, "Optimized feature subsets for epileptic seizure prediction studies," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2011, pp. 1636–1639.

[58] G. Jones, P. Willett, R. C. Glen, A. R. Leach, and R. Taylor, "Development and validation of a genetic algorithm for flexible docking," *Journal of molecular biology*, vol. 267, no. 3, pp. 727–748, 1997.

[59] P. Ataee, A. Yazdani, S. Setarehdan, and H. Noubari, "Genetic algorithm for selection of best feature and window length for a discriminate pre-seizure and normal state classification," in *2007 5th International Symposium on Image and Signal Processing and Analysis*. IEEE, 2007, pp. 107–112.

[60] E. B. Assi, M. Sawan, D. Nguyen, and S. Rihana, "A hybrid mrmr-genetic based selection method for the prediction of epileptic seizures," in *2015 IEEE Biomedical Circuits and Systems Conference (BioCAS)*. IEEE, 2015, pp. 1–4.

[61] T. N. Alotaiby, S. A. Alshebeili, F. M. Alotaibi, and S. R. Alrshoud, "Epileptic seizure prediction using csp and lda for scalp eeg signals," *Computational intelligence and neuroscience*, vol. 2017, 2017.

[62] V. Ganganwar, "An overview of classification algorithms for imbalanced datasets," *International Journal of Emerging Technology and Advanced Engineering*, vol. 2, no. 4, pp. 42–47, 2012.

[63] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015.

[64] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. New York, NY, USA: ACM, 2016, pp. 785–794. [Online]. Available: http://doi.acm.org/10.1145/2939672.2939785

[65] J. A. Suykens, "Nonlinear modelling and support vector machines," in *IMTC 2001. Proceedings of the 18th IEEE Instrumentation and Measurement Technology Conference. Rediscovering Measurement in the Age of Informatics (Cat. No. 01CH 37188)*, vol. 1. IEEE, 2001, pp. 287–294.

[66] F. Samie, S. Paul, L. Bauer, and J. Henkel, "Highly efficient and accurate seizure prediction on constrained iot devices," in *2018 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2018, pp. 955–960.

[67] F. Rosenblatt, "The perceptron: a probabilistic model for information storage and organization in the brain." *Psychological review*, vol. 65, no. 6, p. 386, 1958.

[68] R. P. Costa, P. Oliveira, G. Rodrigues, B. Leitao, and A. Dourado, "Epileptic seizure classification using neural networks with 14 features," in *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*. Springer, 2008, pp. 281–288.

[69] Y. Bengio, "Practical recommendations for gradient-based training of deep architectures," in *Neural networks: Tricks of the trade*. Springer, 2012, pp. 437–478.

[70] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[71] M.-P. Hosseini, H. Soltanian-Zadeh, K. Elisevich, and D. Pompili, "Cloud-based deep learning of big eeg data for epileptic seizure prediction," in *2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE, 2016, pp. 1151–1155.

[72] I. Kiral-Kornek, S. Roy, E. Nurse, B. Mashford, P. Karoly, T. Carroll, D. Payne, S. Saha, S. Baldassano, T. O'Brien *et al.*, "Epileptic seizure prediction using big data and deep learning: toward a mobile system," *EBioMedicine*, vol. 27, pp. 103–111, 2018.

[73] I. Korshunova, "Epileptic seizure prediction using deep leaning," Master's thesis, Ghent University, Belgium, 2014.

[74] Y. LeCun, Y. Bengio *et al.*, "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, p. 1995, 1995.

[75] K. Lehnertz and B. Litt, "The first international collaborative workshop on seizure prediction: summary and data description," *Clinical neurophysiology*, vol. 116, no. 3, pp. 493–505, 2005.

[76] C. Teixeira, B. Direito, M. Bandarabadi, and A. Dourado, "Output regularization of svm seizure predictors: Kalman filter versus the "firing power" method," in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE, 2012, pp. 6530–6533.

[77] G. Welch, G. Bishop *et al.*, "An introduction to the kalman filter," 1995.

[78] L. Chisci, A. Mavino, G. Perferi, M. Sciandrone, C. Anile, G. Colicchio, and F. Fuggetta, "Real-time epileptic seizure prediction using ar models and support vector machines," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 5, pp. 1124–1132, 2010.

[79] A. Chamseddine and M. Sawan, "Deep learning based method for output regularization of the seizure prediction classifier," in *2018 IEEE Life Sciences Conference (LSC)*. IEEE, 2018, pp. 118–121.

[80] B. H. Brinkmann, J. Wagenaar, D. Abbot, P. Adkins, S. C. Bosshard, M. Chen, Q. M. Tieng, J. He, F. Muñoz-Almaraz, P. Botella-Rocamora *et al.*, "Crowdsourcing reproducible seizure forecasting in human and canine epilepsy," *Brain*, vol. 139, no. 6, pp. 1713–1722, 2016.

[81] T. M. Mitchell, "Does machine learning really work?" *AI magazine*, vol. 18, no. 3, p. 11, 1997.

[82] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *International conference on machine learning*, 2013, pp. 1310–1318.

[83] E. Vorontsov, C. Trabelsi, S. Kadoury, and C. Pal, "On orthogonality and learning recurrent networks with long term dependencies," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 3570–3578.

[84] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with lstm," 1999.

[85] M. R. Rajamani, "Data-based techniques to improve state estimation in model predictive control," Ph.D. dissertation, Citeseer, 2007.

[86] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[87] T. Tieleman and G. Hinton, "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude," *COURSERA: Neural networks for machine learning*, vol. 4, no. 2, pp. 26–31, 2012.

[88] M. Riedmiller and I. Rprop, "Rprop-description and implementation details," 1994.

[89] G. E. Dahl, T. N. Sainath, and G. E. Hinton, "Improving deep neural networks for lvcsr using rectified linear units and dropout," in *2013 IEEE international conference on acoustics, speech and signal processing*. IEEE, 2013, pp. 8609–8613.